INTERNET-TECHNOLOGIE
(Zusammenfassung)

ISO	-	International Organisation for Standards.
OSI	-	Open Systems Interconnection $\sim$ Kommunikation offener Systeme.
WWW	-	World Wide Web ('90)
URL	-	Uniform Resource Locator
TCP	-	Transmission Control Protocol (Standardisiert 1983)
NCP	-	Network Control Protocol (ARPANET)
UDP	-	User Datagramm Protocol
IP	-	Internet Protocol (1983)
ICMP	-	Internet Control Message Protocol
IGMP	-	Internet Group Multicast Protokoll
HTTP	-	HyperText Transfer Protocol (Web)
FTP	-	File Transfer Protocol (Remote Login) (1971)
SMTP	-	Simple Mail Transfer Protocol (1972 E-Mail))
MIME	-	Multipurpose Internet Mail Extensions
POP3	-	Post Office Protocol Version 3
IMAP	-	Internet Mail Access Protocol
DNS	-	Domain Name System
PDU	-	Protocol Data Units
PPP	-	Point-to-Point Protocol
MAC	-	Media Access Control
ACK	-	Acknowledgments
RTT	-	Round-Trip-Time
MSS	-	Maximum Segment Size
ICP	-	Internet Caching Protocol
RIP	-	Routing Information Protocol
OSPF	-	Open Shortest Path First
EIGRP	-	Enhanced Interior Gateway Routing Protocol (Cisco)
MTT	-	Maximum Transfer Unit
CDPD	-	Cellular Digital Packet Data
ARP	-	Address Resolution Protocol
FDM	-	Frequency-Division-Multiplexing $\simeq$ Frequenzmultiplexen
TDM	-	Time-Division-Multiplexing $\simeq$ Zeitmultiplexen
MAP	-	Multiple Access Protocol $\simeq$ Mehrfachzugriff
HTML	-	Hyper Text Markup Language
QoS	-	Quality-of-Service
HOL	-	Head-of-the-Line Blocking
HFC	-	Hybrid Fiber Coaxial Cabel
RFC	-	Request for Comments
RTCP	-	Real-Time Control Protokoll
RTSP	-	Real-Time Streaming Protokoll
RSVP	-	Resource Reservation Protokoll
ADSL	-	Asymmetric Digital Subscriber Line
LAN	-	Local Area Network
ATM	-	Asynchronous Transfer Mode

Begriffe

ISO/OSI-Referenzmodell ist ein Modell zur Strukturierung der Kommunikation zwischen Rechner. Um den Nachteil von herstellerspezifischen Produkten zu vermeiden hat ISO Normen in der Netzerkwelt festgelegt. Das **OSI Referenzmodell** legt den Aufbau für alle Arten von Computernetzwerken fest. Die unteren vier Schichten bilden das sogenannte Transportsystem. Ihre Aufgabe ist die Übertragung der Nachricht vom Absender zum Empfänger.

1. Application Layer $\simeq$ Anwendungsschicht **(Nachricht)**
2. Presentation Layer $\simeq$ Darstellungsschicht (Protokoll: NetBIOS)
3. Session Layer $\simeq$ Sitzungsschicht (Beispiel: RPC)
4. Transport Layer $\simeq$ Transportschicht **(Message)**
5. Network Layer $\simeq$ Netzwerkschicht **(Datagramm)**
6. Logical Link Layer $\simeq$ Sicherungsschicht **(Rahmen)**
7. Physical Layer $\simeq$ Physikalischeschicht oder Bitübertragungsschicht

Host teilt sich in Client und Server
Client $\simeq$ Desktop-PCs oder Workstation
Server $\simeq$ Leistungsstarke Maschinen
Store-and-Forward $\simeq$ Speichervermittlung
Best-Effort $\simeq$ Bestem Bemühen
Socket $\simeq$ Tür zwischen zwei Schichten.
Delay $\simeq$ Verzögerung
Congestion $\simeq$ zuverlässig
Broadcast $\simeq$ rundsenden
Verbindungsleitungen Q
Übertragungsrate (Bandbreite) R (bps)
Paketlänge L
CongWin $\simeq$ Überlastfenster
RcvWin $\simeq$ Empfangsfenster.
RcvBuffer $\simeq$ Empfangsbuffer
LastByteAcked $\simeq$ letztes bestätigtes Byte
LastByteSent $\simeq$ letztes gesendetes Bytes
LastByteRead $\simeq$ letztes gelesenes Bytes
LastByteRcvd $\simeq$ letztes empfangendes Bytes
Verkürzungen: $\oplus \cong$ Vorteil; $\ominus \cong$ Nachteil; $\odot \cong$ Eigenschaft

1 Netzwerkkern

Es gibt zwei Vermittlungstechniken zwischen kommunizierenden Endsysteme:

1. Packet-Switching (Paketvermittlung)

▷ Das Internet ist im Wesentlichen ein paketvermitteltes Netzwerk.
▷ erste veröffentliche Arbeit über Paketvermittlung war von Kleinrock 1961
▷ keine Reservierung der Ressourcen (Bandbreite, Puffer).
⇒ mögliche Warteschlangen!
▷ meisten Packet-Switches nutzen Store-and-Forward-Übertragung an den Eingängen der Verbindungsleitungen. Store-and-Forward bedeutet, dass der Switch zuerst das gesamte Paket empfangen muss, bevor er damit beginnen kann, das erste Bit des Pakets über die abgehende Verbindungsleitung zu versenden.
▷ Übertragung von einem Host zu den anderen Host beträgt: $Q \cdot \frac{L}{R}$
⊕ Nachrichten werden in kleinere Paketen geteilt.
⊕ Parallele Übertragung (Knoten können parallel übertragen).
⊕ Bessere gemeinsame Nutzung der Bandbreite als Circuit-Switching
⊕ Implementierung: einfach, effizienter und preisgünstiger.
⊕ Pakete können in verschiedenen Routen verlaufen.
⊕ Geringere Ende-zu-Ende Verzögerung als Message Switching.
⊕ Wenn Bitfehler entstehen wird ein Paket verworfen!
⊖ Header Overhead pro Datenbyte (wegen Steuerinformationen im Paket-Header).
⊖ Wegen Verzögerung sind für die Echtzeitdienste nicht einsetzbar.

Routing-Techniken

1. **Virtueller Kanal-Netzwerke** (Virtueller Kanalnummer) besteht aus:
   (a) einem Pfad zwischen Quell- und Zielhost
   (b) virtuellen Kanalnummer
   (c) Einträgen in VC-Nummerübersetzungstabellen

   - Verbindungsorientierter Dienst auf der Vermittlungsschicht
   - Der ATM Netzwerke bietet nur VC-Dienste.
   - für die laufende Verbindung werden Zustandsinformationen geführt.
   - verhält sich ähnlich wie ein Telefonnetz
   - Der Weg ist festgelegt.
   - Bei einem VC gibt es drei identifizierbare Phasen:
     1. VC-Setup: Während der Setup-Phase kontaktiert der Sender die Vermittlungsschicht, spezifiziert die Empfangsadresse und wartet, bis das Netzwerk den VC einrichtet. Die Vermittlungsschicht bestimmt den Pfad zwischen den Sender und Empfänger und reserviert Ressourcen (z.B. Bandbreite) auf dem Pfad

2. Datentransfer: Nachdem der VC aufgebaut wurde, können Daten darüber fließen.
3. VC Abbau: Wenn der Sender oder Empfänger die Vermittlungsschicht informiert, dass er den VC beenden möchte.

- Der Unterschied zwischen der VC-Aufbau der Vermittlungsschicht und dem Verbindungsaufbau auf der Trasportschicht liegt daran, dass die Packet-Switches auf dem Pfad zwischen den beiden Endsystemen am VC-Setup beteiligt und jeder Paket-Switch hat volle Kenntnis über alle durch ihn durchführenden VCs.

2. **Datagramm Netzwerke** (Hostzieladressen im Header)
   - Verbindungsloser Dienst auf der Vermittlungsschicht
   - führen kein Zustandsinformationen.
   - Der Weg kann während des Prozess geändert werden.
   - Das Internet bietet der Transportschicht nur einen Datagramm-Dienst.
   - Bei der **Datagramm-Vermittlungsschicht** muss ein Endsystem jedes Mal, wenn es ein Packet senden will, das Paket mit der Adresse des Zielendsystems versehen und dann das Paket in das Netzwerk einspeisen.
   - Analogie: Fahren mit dem Auto und in jede Kreuzung anhalten und fragen.

Die zwei Arten von Packet Switching:

- **Bridges** (Schicht $1-2$)
  ▷ In kleineren Netzwerken von paar hunderte Hosts werden Bridges eingesetzt. Man braucht auch keine IP-Adressen zu konfigurieren.
  ⊕ Verbinden Ethernet-Segmente unterschiedlicher LAN-Technologien, darunter 10 Mbps und 100 Mbps Ethernet.
  ⊕ Besteht keine Beschränkung hinsichtlich der Größe der LAN.
  ⊕ Sind Plug-and-Play Geräte
  ⊖ Verwenden Spanning-Tree Protokoll

- **Router** (Schicht $1-3$)
  ▷ Bei größere Netzwerken werden zusätzlich zu Bridges auch Router eingesetzt.
  ⊕ Wegen ein spezielles Feld im IP-Datagramm-Header kreisen die Pakete nicht durch den Router.
  ⊕ Router verwenden Datagramme
  ⊕ Sie können den besten Pfad zwischen Host und Ziel wählen
  ⊕ Router bieten Firewall-Schutz vor Broadcast-Flutungen
  ⊖ Router haben kein Plug-and-Play Fähigkeit.
  ⊖ Router haben oft längere Verarbeitungszeiten pro Paket als Bridges.
  ⊖ Router sind Zustandlos.

1.1 Message-Switching (Nachrichtenvermittlung)

▷ Spezifischer Art von Packet-Switching.
▷ Nachrichten werden als ein ganzer Packet geschickt (kein Segmentierung).
⊖ Sequentielle Übertragung (wenn ein Knoten überträgt andere warten)
⊖ Wenn Bitfehler enstehen wird ganzer Nachricht verworfen.

2. Circuit-Switching (Leitungsvermittlung)

▷ Das Netzwerk baut einen dedizierten Ende-zu-Ende Schaltkreis zwischen zwei Hosts auf. Diese Technik wird in Telefonnetzwerke benutzt.
▷ Reservierung von Ressourcen für die Dauer der Sitzung
▷ Konstante Rate wird für die Dauer der Verbindung reserviert
⇒ möglicherweise unnötige Reservierung.
▷ Für die gesamten Dauer der Verbindung wird der Verbindungszustand geführt
⊖ Alle Pakete verlaufen auf der gleichen Route.
▷ Jede Verbindungsleitung hat n Schaltkreisen
▷ Ein Schaltkreis auf einer Verbindungsleitung wird entweder durch: FDM oder TDM implementiert. Bei **FDM** erhält jeder Schaltkreis kontinuierlich einen Anteil an der Bandbreite. Bei **TDM** erhält jeder Schaltkreis periodisch in kurzen Zeitintervallen die gesamte Bandbreite.
⊖ komplexe Signalisierungsoftware voraussetzt.

Ein **Protokoll** definiert das Format und die Reihenfolge von Nachrichten, die zwischen zwei oder mehr kommunizierenden Endsysteme ausgetauscht werden, sowie die Handlungen, die bei der Übertragung beim Empfang einer Nachricht oder eines anderen Ereignisses unternommen werden.

Mit **Dienst** (Service) wird definiert, was die Schicht macht und nicht wie die darüberliegenden Einheiten darauf zugreifen oder wie die Schicht funktioniert.

Zugangsnetzwerke

- **Private Endsysteme:**
  Modem (56 Kbps), ISDN (Vollständig digital 128 Kbps), ADSL (Downstream-, Upstream-Kanal) und HFC-Netzwerk (anhand Kabelnetzes).
- **Institutionelle Endsysteme:**
  LAN-Technologie ($10 - 100$ Mbps und 1Gbps)
- **Mobile Endsysteme:**
  Funkspektrum (durch ein virtuelles Netzwerk CDPD) mit einer Basisstation; KBit Bereich; MAC-Protocol

Physikalische Medien

– Es gibt zwei Arten von Übertragungsmedien:

1. **geführte**: die Wellen werden an einem festem Medium wie Glasfaser-, Kupfer- und Koaxialkabel entlanggelenkt.
2. **ungeführte** die Wellen breiten sich in der Atmosphäre und im Raum aus CDPD-System oder Satellitenkanal.

- **Kupferdoppelader (Twisted-Pair):** billig, häufig benutzte
- **Unabgeschirmtes verdrilltes Kabelpaar (UTP):** LAN
- **Koaxialkabel:** Basisband (50-Ohm-Kabel), Breitband (75-Ohm-Kabel)
- **Glasfaser:**
  Daten werden durch Licht übertragen, wobei jedes Impuls ein Bit darstellt.
- **Erdgebundene und Satellitenkanäle, GEO- und LEO- Satelliten**

Delays (Verzögerungen)

Die Gesamtverzögerung oder **Ende-zu-Ende Verzögerung** ist die Summe der :

- **Verarbeitungsverzögerung:** Die Zeit, die benötigt wird für die Feststellung, wohin das Paket weiterzuleiten ist, um das Paket auf Bitfehler zu prüfen und für die Durchsicht des Paket-Headers.
- **Warteschlangenverzögerung:** Die Warteschlangen in einem Router sind genau die Stellen, wo Pakete verloren gehen oder verworfen werden.
  Verkehrsintensität ist $\frac{a \cdot L}{R}$, wobei a die Durchnittsrate ist.
- **Übertragungsverzögerung:** Die Zeit, die der Router benötigt, um das Paket abzuschicken. (wird auch als Store-and-Forward-Verzögerung bezeichnet) $\frac{L}{R}$
  Übertragungsverzögerung hat nichts mit der Entfernung zwischen Routern zu tun. Übertragungsverzögerung ist eine Funktion von L und R und nicht der Entfernung zwischen zwei Routern
- **Ausbreitungsverzögerung** ist die Zeit, die es dauert, bis ein Bit sich von einem Router zum nächsten ausbreitet. Die Ausbreitungsverzögerung ist die Entfernung zwischen zwei Routern, geteilt durch die Ausbreitungsgeschwindigkeit. Ist ein Funktion der Entfernung zwischen den zwei Routern, hat aber nichts mit L und R zu tun

Eine wichtige Komponente der Ende-zu-Ende-Verzögerung sind die zufälligen Warteschlangenverzögerungen in den Routern. Aufgrund dieser schwankenden Verzögerungen im Netzwerk kann die Zeit zwischen der Erzeugung eines Pakets in der Quelle und der Ankunft beim Empfänger von einem Paket zum nächsten schwanken. Dieses Phänomen

wird als **Jitter** bezeichnet.

Bandbreiten-sensitive Anwendungen setzen ein bestimmte Bandbreite voraus. (Multimedia Anwendungen)

Elastische Anwendungen nutzen so viel oder so wenig Bandbreite wie momentan zur Verfügung steht. (Email, Web, FTP).

Funktionen der Layers (Schichten)

⊕ Jede Schicht erfüllt genau eine definierte Funktion.
⊕ Der Nachrichtentransport wird in kleinere Teilaufgaben zerlegt.
⇒ Komplexität wird verteilt und somit verringert.
⊕ Im Falle von Protokolländerungen innerhalb einer Ebene bleiben die übrigen Protokolle davon unverändert.
⊖ Eine Schicht kann vielleicht die gleiche Funktionalität wie ein niedriger Schicht ausführen.
⊖ dass die Funktionalität auf einer Schicht möglicherweise Informationen benötigt, der nur auf einer anderen Schicht vorhanden ist; dies verletzt die Trennung von Schichten.

- Flusskontrolle: Wegen Überschwemmung
- Fehlerkontrolle: Zuverlässigkeit
- Segmentierung und Reassemblierung
- Multiplexen
- Verbindungsaufbau: Für das Handshake mit einem Kommunikationspartner

ISO/OSI-Schicht

Die **Anwendungsschicht** ist zuständig für die Unterstützung von Netzwerkanwendungen. Sie beinhaltet Protokolle wie HTTP, FTP, SMTP.
Ein **Socket** ist die Schnittstelle zwischen der Anwedungs- und der Transportschicht.

Präsentationsschicht kümmert sich mehr um Syntax und Semantik der übertragenen Informationen. Ein typisches Beispiel für ein Dienst der Präsentationsschicht ist die Kodierung und Dekodierung von Daten auf standardisierte und vereinbarte Weise.

Die **Sitzungsschicht** ermöglicht den gewöhnlichen Datentransport, wie die Transportschicht auch, bietet aber zusätzlich erweiterte Dienste, die für bestimmte Anwendungen nützlich sind. Ein Benutzer kann sich z.B. in einer Sitzung an einem entfernten System anmelden oder Dateien zwischen zwei Maschinen übertragen. Weitere spezielle Diente: Dialogsteuerung, Token-Managment, Synchronisation usw.

Transportschicht übernimmt den Transport von Nachrichten der Anwendungsschicht zwischen Client und Server. (TCP, UDP). Transportschicht teilt eine Nachricht ggf. in

kleinere Einheiten für die Vermittlungsschicht auf und überwacht den Datentransport zwischen Sender und Empfänger.

Vermittlungsschicht ist zuständig für die Weiterleitung von Datagrammen (Wegwahl) von einem Host zum anderen.

Die **Sicherungsschicht** leitet ein Paket durch eine Reihe von Packet-Switches zwischen der Quelle und dem Ziel weiter. (Ethernet, PPP). Die Sicherungsschicht ist dafür verantwortlich, dass die Daten genau in der Reihenfolge ankommen, in der sie auf der Senderseite abgeschickt wurden.

Die **Bitübertragungsschicht** ist für das Übertragen von Bitströmen verantwortlich. Bitübertragungsschicht hat die Aufgabe die einzelnen Bits im Rahmen von einem Knoten zum nächsten zu übertragen.

2 Anwendungsschicht

Ein Protokoll der Anwendungsschicht ist nur ein Teil (allerdings größer) Teil einer Netzwerkanwendung. Alle Netzwerkanwedungsprotokolle haben zwei Seiten: eine Client- und eine Server-Seite.

Die **Web-Anwendung** besteht aus vielen Komponenten, darunter Web Browser, HTML, Web-Server und HTTP.

HTTP definiert die Art der Weiterleitung von Nachrichten zw. Browser und Web-Server.

HTTP wird in zwei Programmen implementiert: in einem Client und einem Server-Programm.

- HTTP ist ein Datentransferprotokoll, das auf TCP setzt.
- HTTP muss sich nicht um verlorene Daten oder um die Details kümmern.
- HTTP ist ein Zustandsloses Protokoll (ohne Zustandsinformationen).
- HTTP ist ein **Pull-Protokoll**.
- HTTP unterstützt sowohl persistente als auch nicht persistente Verbindungen.
- HTTP hat zwei Versionen, die miteinander kompatible sind:

    1. **HTTP 1.0** (bis 1997), die mit eine nicht persistente Verbindung arbeitet.
       • Nachdem der Server das Objekt geschickt hat, wird jede TCP-Verbindung geschlossen.
       • Eine TCP-Verbindung befördert genau eine Anfrage- und eine Antwortnachricht.

⊖ Für jedes Objekt wird eine neue Verbindung angefordert. Für jede Verbindung müssen TCP-Puffer und TCP-Variablen zugeteilt.
⊖ Jedes Objekt muss zwei RTT erdulden.
⊖ Slow-Start (Verbesserung: parallele TCP-Verbindungen)

2. **HTTP 1.1** (ab 1998), die mit eine persistente Verbindung arbeitet.
   • Der Server lässt die TCP-Verbindung offen, nachdem er eine Antwort gesendet hat.
   • Zwei Versionen: ohne Pipelining und mit Pipelining.
   • Beide Versionen haben eine kürzere Slow-Start-Verzögerung. Nach dem Versenden des ersten Objekts das nächste Objekt nicht in der anfänglichen langsamen Rate senden muss, weil er weiterhin die gleiche TCP-Verbindung benutzt
   • **ohne Pipelining:** Der Client gibt nur dann eine neue Anfrage aus, wenn die vorherige Antwort nicht empfangen wurde. Es wird ein RTT pro Objekt benötigt. Nachteil dieser Version ist, dass die Verbindung hängt (untätig ist), nachdem der Server ein Objekt über persistente TCP-Verbindung gesendet hat und auf die Auskunft einer weiteren Anfrage wartet. Diese "Hängen" verschwendet Server-Ressourcen.
   • **mit Pipelining:** Der Client gibt eine Anfrage aus, sobald er auf eine Referenz stößt. TCP-Verbindung hängt weniger lang als ohne Pipelining. Werden alle Anfragen und alle Antworten aufeinander folgend gesendet, dann wird nur ein RTT benötigt.

- Es gibt zwei HTTP-Nachrichtentypen:
  • **Anfragenachricht** besteht aus 5 Zeilen (Erste Anfragezeile und der Rest Header-Zeilen)
  • **Antwortnachricht** umfasst drei Abschnitte: eine Statuszeile, sechs Header-Zeilen und einen Entity Body.

- HTTP bietet zwei Mechanismen für Identifizierung von Benutzern:

  1. **Authentifizierung** (Benutzernamen und Passwort)
  2. **Cookies** sind Text-Dateien, die vom Server auf dem Rechner des Clienten gespeichert werden. Wenn ein Client eine Web-Seite besucht (Verbindungsanfrage stellt), generiert der Server einen zufälligen Nummer und behält sie im Erinnerung. Wenn die Seite wieder aufgerufen wird, dann kennt der Server zwar nicht den Benutzernamen, er weiß aber, dass es sich um den gleichen Benutzer handelt, der zuvor eine spezifische Anfrage geschickt hat. **Gründe:** 1. Kein Aufforderung für die Eingabe von Benutzernamen und Passwort. 2. Werbung und Statistiken für den Server 3. Beim Online-Einkauf die Liste zu verfolgen.

RTT ist die Zeit, die ein kleines Paket braucht, um vom Client zum Server und wieder als Bestätigung zurück zum Client zu reisen. Die RTT beinhaltet Ausbreitungsverzögerungen, Warteschlangenverzögerungen in dazwischen liegenden Routern und Switches

sowie Verarbeitungsverzögerungen

Die **gesamte Reaktionszeit**, d.h. die Zeit ab der Anfrage des Browsers für ein Objekt bis zum Empfang des Objekts, ist die Summe der LAN-Verzögerung, der Zugangsverzögerung und der Internet-Verzögerung.

Web-Caches, auch Proxy-Server genannt, werden sehr häufig verwendet.

- Die Reaktionszeit auf eine Client-Anfrage kann erheblich reduziert werden.
- Der Verkehr zwischen Institutionen erheblich reduzieren.
- Innerhalb der Institutionen wird eine schnelle Verteilung von Inhalten geben.

ICP ist ein Protokoll der Anwendungsschicht, das es einem Cache ermöglicht, schnell bei einem anderen Cache um ein bestimmtes Dokument anzufragen.

FTP ist ein Protokoll der Remote Login für die Übertragung einer Datei von einem Host zu einem anderen.

Damit der Benutzer Zugang zu dem entfernten System erhält, muss er eine Benutzeridentifizierung und ein Passwort eingeben.

FTP benutzt zwei parallele TCP-Verbindungen, um eine Datei zu übertragen

1. **Steuerverbindung:** Steuerinformationen zwischen zwei Hosts auszutauschen. Während der gesamten Dauer einer Sitzung bleibt die Steuerverbindung offen.
2. **Datenverbindung:** für die Übertragung der Dateien. Für jede übertragene Datei innerhalb einer Sitzung wird eine Datenverbindung aufgebaut.

HTTP und TCP sind Datentransferprotokolle und beide setzen auf TCP auf.

Im Verlauf einer Sitzung muss der FTP-Server den Zustand über die Benutzersitzung führen.

Da FTP eine getrennte Steuerverbindung benutzt, sagt man es sendet seine Steuerinformationen Out-of-Band. HTTP und SMTP senden seine Steuerinformationen In-Band.

Das führen von zustandsinformationene schränkt die Anzahl der Sitzungen, die FTP gleichzeitig unterstützen kann.

E-Mail-Anwendung besteht aus einem Mail-Server, Mail-Reader (User-Agents) und SMTP. Ein *User-Agent* ist eine Schnittstelle zwischen dem Benutzer und der Netzwerkanwendung. SMTP überträgt Nachrichten von dem Mail-Servern der Sender zu denjenigen der Empfänger.

- SMTP ist eine alte Technologie

- Beschränkt es den Rumpf aller Mail auf einfaches 7-Bit ASCII. Dieser Einschränkung war 80'er sinnvoll als die Übertragungskapazität knapp war und ohne die riesigen Attachments. **Nachteil** Die binären Multimedia-Daten müssen zuerst in ASCII-Code kodiert werden und am Ende umgekehrt.
- SMTP normalerweise beim Versenden von Mail keine dazwischen liegenden Mail-Server benutzt.

Unterschiede zu HTTP:

- beide verwenden TCP-Verbindungen und werden für die Übertragung von Dateien von Host an einem anderen benutzt.
- SMTP überträgt Dateien (in diesem Fall E-Mails) von einem Mail-Server zu einem anderen.
- HTTP überträgt Dateien vom Web-Server zum Browser.
- beide verwenden persistente Verbindungen
- HTTP ist ein Pull-Protokoll (zieht) und SMTP ist ein **Push Protokoll** (schiebt)
- Kodierung: SMTP 7-Bit-ASCII Format HTTP (weder persistentes als auch nicht persistentes) setzen kein ASCII-Kodierung voraus.
- Das Web kapselt jedes Objekt in einer getrennten HTTP-Antwortnachricht sendet. E-Mail stellt demgegenüber alle Objekte in die gleiche Nachricht.

▷ Ein E-Mail Nachricht enthält ein Header und ein Rumpf.
▷ **MIME** ist eine Erweiterung von E-Mail Informationen. MIME enthält zwei wichtige Header für die Unterstützung von Multimedia-Dateien

- Content-Type: Typ der Multimedia-Datei
- Content-Transfer-Encoding: Die verwendete Kodier-Methode.

Zugangsprotokolle

Derzeit gibt es drei Mail-Zugangsprotokolle:

1. **POP3** (Port 110)
   - extrem einfaches Mail-Zugangsprotokoll
   - Funktionalität begrenzt
   - POP3 führt Zustandsinformationen nur innerhalb der Pop3-Sitzung

- besteht aus drei Phasen
  1. Autorisation Benutzernamen und Passwort
  2. Transaktion Der User-Agent kann Nachrichten "*herunterladen und löschen*" oder "*herunterladen und behalten*" und Mail-Statistiken abrufen
  3. Aktualisierung nach QUIT Befehl wird der POP3-Sitzung beendet und alle markierten Nachrichten werden vom Mail-Server gelöscht.

2. **IMAP**

   - mehr Eigenschaften aber auch komplexer als POP3
   - Ermöglicht dem Benutzer die Manipulation entfernter Mailboxen, als wären sie auf dem lokalen Computer vorhanden.
   - IMAP verwaltet für jeden seiner Benutzer eine Ordner-Hierarchie.
   - bietet Befehle, mit denen ein User-Agent nur den Nachrichten-Header einer Nachricht oder nur einen Teil einer MIME-Multipart Nachricht abrufen kann. Vorteil bei langsamen Verbindungen und PC mit wenig Speicher.
   - IMAP-Server befindet sich immer in einem von vier Zuständen:
     1. Non-authenticated States: Benutzernamen und Passwort
     2. Authenticated States Ordner auswählen
     3. Selected States Befehle ausgeben
     4. Logout States wird die Sitzung beendet

3. **HTTP Web-basiert E-Mail**

   - Hotmail, Yahoo
   - sehr beliebt aber langsam

DNS

Wichtig (Seite 139): Was passiert, wenn der Client ein URL eingibt?

Ein Host lässt sich auf zweierlei Art identifizieren: durch einen Hostnamen oder ein IP-Adresse.

DNS ist **(1)** eine verteilte, in einer Hierarchie von Name-Servern implementierte Datenbank und **(2)** ein Protokoll der Anwendungsschicht, das es Host und Namen-Server ermöglicht, im Rahmen des Namendienstes zu kommunizieren.

- DNS ist kein Anwendung, mit der der Benutzer direkt interagieren kann.
- DNS ist ein Verzeichnisdienst, der Hostsnamen in IP-Adressen übersetzt.
- DNS wird üblicherweise von anderen Protokolle wie HTTP, SMTP, FTP usw. benutzt.
- DNS setzt auf UDP auf und benutzt Port 53

- DNS ist für Internet-Anwendungen eine zusätzliche Verzögerung. (Verbesserungsmöglichkeit durch Cache)

- DNS bietet außerdem
  1. Host-Aliasing: Ein Host mit einem komplizierten Hostnamen kann einen oder mehrere Aliasnamen haben.
  2. Mail-Server-Aliasing Email Adressen sollten mnemonisch sein.
  3. Lastverteilung Wenn die Seiten stark frequentiert sind dann werden auf mehrere Servern repliziert.

Da kein einzelner Namen-Server alle Übersetzungen für alle Hosts im Internet enthält (Wegen: Ausfall einem einzigen Name-Server, riesige Datenbank, Verkehrsvolummen) gibt es drei Arten:

- Der **Lokale Name-Server** befindet sich normalerweise in der Nähe des Client (gleicher LAN oder innerhalb der ISP)

- **Root Name-Server** Wenn der Lokale Name-Server nicht beantworten kann, dann richtet sich bei Root Server-Name. Möglicherweise verfügt der angefragte Root Name-Server nicht über einen Eintrag des gesuchten Hostnamens. Er kennt aber die IP-Adresse eines autoritativen Name-Server. Die Anfrage geschieht auf Rekursive- oder Iterative-Weise. Der größte Teil von Root Name-Server sind in Nord-Amerika.

- **Autoritative Name-Server** Jeder Host ist bei einem autoritativen Name-Server registriert. Eigentlich muss jeder Host mindestens zwei autoritative Name-Server haben für den Fall, dass einer ausfällt.

$\oplus$ Um die Verzögerungen zu verkürzen und die Anzahl von DNS-Nachrichten im Netzwerk zu reduzieren benutzt man DNS-Caching.

Die Name-Server speichern **R**esource-**R**ecords für die Übersetzung von Hostnamen in IP-Adressen. Jede DNS-Antwortnachricht enthält einen oder mehrere RR. Ein RR ist ein 4-Tupel (*Name, Wert, Typ, TTL*)

Es gibt zwei DNS-Nachrichtentypen: DNS-Anfrage- und DNS-Antwortnachrichten.

3 Internet-Transportprotokolle und ihre Dienste

Ein Protokoll der Transportschicht bietet eine logische Kommunikation (also nicht physische) zwischen Anwendungsprozessen, die auf unterschiedliche Hosts laufen.

Anwendungsprozesse verwenden die von der Transportschicht bereitgestellte logische Kommunikation, um Nachrichten miteinander auszutauschen, ohne sich um die Details der physikalischen Infrastruktur, über die diese Nachrichten fließen, kümmern zu müssen.

Die grundlegende Verantwortung von UDP und TCP ist im Wesentlichen die Erweiterung des IP-Übertragungsdienstes zwischen zwei Endsystemen auf ein Übertragungsdienst zwischen zwei Prozessen, die auf zwei Endsystemen laufen. Diese Erweiterung nennt man Anwendungsmultiplexen und -demultiplexen.

Die Aufgabe der Übertragung der in einem Transportschichtsegment enthaltenen Daten an den richtigen Anwendungsprozess nennt man **Demultiplexen**

Die Aufgabe des Einsammelns von Daten im Quellhost aus verschiedenen Anwendungsprozessen, die Vervollständigung von Daten mit Header-Informationen, um Segmente zu bilden, und die Weiterleitung der Segmente an die Vermittlungsschicht wird **Multiplexen** bezeichnet.

Verbindungsorientierter Dienst: TCP

- TCP wurde im Jahre 1974 entwickelt, bevor es PSs Workstations, Web, LAN-Technologien gab!
- TCP-Verbindung bietet Vollduplex-Datentransfer.
- TCP-Protokoll läuft nur in den Endsystemen und nicht in den dazwischen liegenden Netzwerkelementen (Router und Bridges)
- TCP baut eine logische Verbindung zwischen Sender und Empfänger.
- Ein TCP-Verbindung in beide Richtungen setzt sich zusammen aus Sende- und Empfangspuffer, Variablen und zwei Sockets-Verbindungen.
- Die Fehlerkontrolle wird nicht für Pakete, sondern für zu übertragende Bytes durchgeführt.
- **Drei-Weg-Handshake:**
  Bevor die beiden Prozesse mit der Kommunikation beginnen gibt es eine (Begrüßung) Austausch von TCP-Segmenten zwischen Client und Server.
  * Die ersten beiden Segmente enthalten keine Nutzdaten und das dritte kann Nutzdaten enthalten. Zuerst sendet der Client eine Verbindungsanfrage (Ein speziellen Segment mit der SYN-Bit auf 1 gesetzt) an dem Sender.
  * Zweitens extrahiert der Server das SYN-Segment aus dem Datagramm, weist

der Verbindung TCP-Puffer und Variablen zu und sendet ein Segment mit drei wichtigen Daten (SYN auf 1, erhöht den Bestätigungsfeld client_isn + 1 und setzt eine eigene Anfangssequenznummer).
∗ Zuletzt beim Empfang des SYNACK-Segments weist der Client seinerseits Verbindung Puffer und Variablen zu. Anschließend sendet der Client-host dem Server noch ein Segment, mit dem er das SYNACK-Segment des Server bestätigt und setzt SYN auf 0.
Nach Handshake-Prozedur ist die TCP-Verbindung zwischen den beiden Hosts aufgebaut und es kann mit der Übertragung der Dateien beginnen.
Drei-Weg-Handshake ermöglicht es dem Sender und Empfänger, die erforderlichen Zustandsinformationen einzurichten. Während der Drei-Weg-Handshake wird ein Puffer vorbereitet. Der Client-Prozess gibt einen Datenstrom durch das Socket an TCP, das auf dem Client läuft und TCP leitet diese Daten in Segmenten an den **Sendepuffer** weiter, wobei man das MSS beachten muss. TCP kapselt jedes Client-Datenstück mit einem TCP-Header und formt damit TCP-Segmente. Die Segmente werden nach unten an die Vermittlungsschicht weitergeleitet. Wenn TCP an anderen Ende ein Segment empfängt, stellt es die Daten des Segments in den **Eingangspuffer** der TCP-Verbindung. Anschließend liest Anwendung den Datenstrom aus des diesem Puffer.

- **zuverlässige Datentransfer:**
  Mit Hilfe von Flusskontrolle, Sequenznummern, Bestätigungen(ACKs) und Timern gewährleistet TCP, dass die Daten irgendwann in die richtige Reihenfolge und ohne Fehler ankommen.
  Der zuverlässige Datentransferdienst stellt sicher dass der Datenstrom, den ein Prozess aus seinem TCP-Empfangspuffer liest, nicht beschädigt ist, keine Lücken hat und nicht dupliziert wurde.

  Über einen absolut zuverlässigen Kanal: rdt1.0
  Der Sender und Empfänger haben jeweils einen Zustand. Hat kein Bestätigungen.

  Über einen Kanal mit Bitfehlern: rdt2.0
  Sender-Seite hat zwei Zustände und Empfänger einen Zustand.
  In diesem Protokoll werden Bitfehler behandelt sowie Neuübertragungen durchgeführt (**A**utomatic-**R**epeat-re**Q**uest-Protokolle). In ARQ-Protokollen sind drei Fähigkeiten, die vorhandene Bitfehler behandeln:
  1. Fehlererkennung
  2. Positive (ACK) und negative (NAK) Bestätigungen je ein Bit lang.
  3. Neuübertragung.
  Wenn der Empfänger in einem Wartezustand auf ein ACK oder NAK befindet, kann keine weiteren Daten von der höheren Schicht empfangen. Aufgrund diese Verhalten bezeichnet man rdt2.0 als Stop-and-Wait Protokoll.
  Nachteil: es können auch ACK oder NAK-Pakete beschädigt werden!
  – **Lösung rdt2.1:** Der Sender nummeriert seine Datenpakete. Der Sender und

Empfänger haben je zwei Zustände.
– **rdt2.2** ist ein NAK-freies zuverlässiges Datentransferprotokoll für einen Kanal mit Bitfehlern "Dreifacher Duplikat-ACKs"

Über einen verlustbehafteten Kanal mit Bitfehlern: rdt3.0
Für die Neuübertragungsmechanismus werden Countdown-Timer implementiert. rdt3.0 wird auch als *Alternating-Bit-Protokoll* genannt.

Zuverlässige Datentransfer mit Pipelining:
Anstatt mit Stop-and-Wait Betrieb zu arbeiten, lässt man den Sender mehrere Pakete senden, ohne dass er auf Bestätigungen warten muss.
Diese Technik hat zwei Konsequenzen:
1. Der Bereich der Sequenznummern muss vergrößert werden
2. Die Sender- und Empfängerseite des Protokolls benötigen einen Puffer für mehr als ein Paket.
Für die Wiederherstellung nach Fehlern mit der Pipelining-Technik sind zwei grundlegende Ansätze bekannt:

1. **Go-Back-N:** Bei einem GBN-Protokoll ist es dem Sender gestattet, mehrere Pakete (sofern vorhanden) zu übertragen, ohne auf eine Bestätigung warten zu müssen. Insgesamt darf er aber nicht als eine bestimmte Höchstzahl N (wegen der Flusskontrolle), an unbestättigten Paketen in die Pipeline geben. N wird meist als Fenstergröße (**Window Size**) und das GBN-Protokoll als **Sliding-Window-Protokoll** bezeichnet. GBN-Protokoll benutzt ACKs, jedoch keine NAKs.
   Nachteil: Wenn die Fenstergröße und das Bandbreite/Verzögerung-Produkt groß sind, können sich viele Pakete in der Pipeline befinden. Ein einzelner Paketfehler kann GBN veranlassen, eine große von Paketen erneut zu übertragen, von denen viele vielleicht unnötig sind.
   Das GBN-Protokoll schaltet für jedes Segment ein Timer.
   Timer-Ereignis: Wenn der Timer abläuft, überträgt der Sender alle Pakete, die zuvor gesendet, aber noch nicht bestätigt wurden, noch einmal.
2. **Selective Repeat** vermeidet unnötige Neuübertragungen. Die selektive Neuübertragung setzt voraus, dass der Empfänger korrekt empfangene Pakete individuell bestätigt.
   Die Fenstergröße von N für SR-Protokolle muss kleiner als oder gleich die Hälfte der Größe des Sequenznummenrraums sein muss.
   Auch hier werden die Timern ähnlich wie bei GBN benutzt aber außer der Reihe ankommende Pakete werden so lange zwischengespeichert, bis eventuell fehlende Paket empfangen wird.

- **Flusskontrolle** ist ein Dienst zur Abstimmung von Geschwindigkeiten.
  Die Flusskontrolle stellt sicher, dass keine Seite einer Verbindung durch zu schnelles Senden zu vieler Pakete überschwemmt. Der Flusskontrolldienst zwingt das

sendende Endsystem seine Rate zu reduzieren, sobald sich ein solches Risiko abzeichnet.
TCP stellt Flusskontrolle dadurch, dass es den Sender eine Variable verwaltet lässt, die man als **Empfangsfenster** (Receive Window) bezeichnet. Informell wird das Empfangsfenster benutzt, um dem Sender eine Vorstellung davon zu vermitteln, wie viel freier Pufferplatz beim Empfänger zur Verfügung steht. Das Empfangsfenster ist dynamisch, d.h. es ändert sich im Verlauf einer Verbindung. (Das Beispiel im Buch ansehen!)

- **Überlastkontrolle** verhindert, dass das Internet in einen Verkehrstau gerät.
  ▷ Ist ein Router überlastet, können seine Puffer überlaufen und Pakete verloren gehen.
  ▷ Das Überlastkontrollmechanismus zwingt jede neue TCP-Verbindung, dass am Anfang in einer relativ langsamen Rate (kleine w) (**Slow-Start**) zu übertragen, die dann aber bis auf eine relativ hohe Rate (größere w) gesteigert werden kann, wenn dass Netzwerk nicht überlastet ist. Die Slow-Start-Phase endet, wenn die Fenstergröße den Wert von Grenzwert übersteigt.
  ▷ Der Überlastkontrollmechanismus von TCP lässt jede Seite der Verbindung zwei zusätzlich Variablen verwalten:
  1. Überlastfenster (Congestion Window) bringt die zusätzliche Einschränkung, wie viel Verkehr ein Host in eine Verbindung einspeisen kann. Insbesondere darf die bei einem Host für eine TCP-Verbindung anstehende unbestätigte Datenmenge des Minimum von CongWin und RcwWin nicht übersteigen

  $$\textbf{LastByteSent} - \textbf{LastByteAcked} \leq \textbf{min\{CongWin, RcvWin\}}$$

  Das TCP-Überlastfenster reguliert die Zeiten, in denen die Segmente an die Vermittlungsschicht weitergeben werden. Anfangs entspricht das Überlastfenster einer MSS. TCP sendet das erste Segment zum Netzwerk und wartet auf eine Bestätigung. Wird dieses Segment bestätigt, bevor sein Timer abläuft, erhöht der Sender das Überlastfenster um eine MSS und sendet zwei Segmente mit maximaler Größe. Wenn diese Segmente vor ihrem jeweiligen Timeout bestätigt werden, erhöht der Sender das Überlastfenster um eine MSS pro bestätigtem Segment, so dass das Überlastfenster jetzt vier MSS umfasst, und sendet vier Segmente mit maximaler Größe. Dieses Prozedur wird so lange fortgesetzt, (1) solange das Überlastfenster unter dem Grenzwert liegt und (2) die Bestätigung vor ihrem jeweiligen Timeout ankommen.
  Fazit:
  – Solange das Überlastfenster unter dem Grenzwert liegt, wächst exponentiell
  – Steigt das Überlastfenster über dem Grenzwert, wächst es linear. Diese Phase wird als **Überlastvermeidung** bezeichnet.
  – Bei jedem Timeout wird der Grenzwert auf die Hälfte des aktuellen Überlastfensters zurückgesetzt und das Überlastfenster auf 1 gesetzt.
  2. Grenzwert (Threshold) ist eine Variable, die sich darauf auswirkt, wie Überlastfenster wächst.

▷ Eine TCP-Verbindung steuert ihre Übertragungsrate durch Einschränkung der Anzahl von übertragenen, jedoch noch nicht bestätigten Segmenten. Die Anzahl von zulässigen unbestättigten Segmenten w wird als *TCP-Fenstergröße* genannt.
▷ Im Falle eines Verlusts reduziert die TCP-Verbindung w auf ein "sicheres Maß" und beginnt anschließend erneut mit dem Abtasten auf eine mögliche unbenutzte Bandbreite, indem sie w langsam wieder erhöht.
▷ Es gibt zwei Arten von Überlastinformationen die vom Netzwerk zum Sender geschickt werden:
1. Direktes Feedback: Choke-Paket
2. Markierung: Es fordert eine volle RTT-Zeit
▷ Eine wichtige Messgröße der Leistung einer TCP-Verbindung ist ihr Durchsatz- die Rate, in der sie Daten vom Sender zum Empfänger überträgt. Der Durchsatz hängt von dem Wert von w ab. Wenn ein TCP-Sender alle w-Segmente nacheinander überträgt, muss er anschließend eine RTT warten, bis er Bestätigungen für diese Segment erhält. an diesem Punk kann er w weitere Segmente senden. Wenn die Verbindung w Segmente mit einer Größe von MSS Byte alle RTT Sekunden überträgt, dann beträgt der Durchsatz der Verbindung bzw. die Übertragungsrate $\frac{(w \cdot MSS)}{RTT}$ Bytes pro sec.
▷ Wenn RTT und W über die Dauer der Verbindung ungefähr konstant sind, liegt die TCP-Übergangsrate im Bereich von $\frac{W \cdot MSS}{2 \cdot RTT}$ bis $\frac{W \cdot MSS}{RTT}$

▷ Wenn man die Slow-Start-Phase ignoriert:
Fall 1: Wenn sein Netzwerkpfad nicht Überlastet ist, dann wächst das Fenster linear (pro RTT um 1 → Additive Increase)
Fall 2: Wenn sein Netzwerkpfad überlastet ist, senkt das Überlastfenster um Faktor 2 pro RTT (Multiplicative Decrease).
▷ Bei einem einzelnen Bottleneck ist TCP fair (siehe Beispiel im Buch!)
Überlastkontroll-Algorithmen:
`1. Tahoe-Algorithmus` siehe das Beispiel im Buch
⊖ Wenn ein Segment verloren geht, muss die Senderseite der Anwendung, eventuell längere Zeit auf ein Timeout warten.
`2. Reno-Algorithmus` ist eine Variante von Tahoe. Reno beinhaltet Fast-Retransmit-Mechanismus (Drei Duplikat ACKs) und Fast-Recovery-Mechanismus. Reno wird in meisten heutigen TCP-Implementierungen verwendet.
`3. Vegas-Algorithmus` versucht die Leistung von Reno zu verbessern. Das Grundkonzept bei Vegas ist, (1) dass man vor einem Paketverlust in den Routern zu erkennen und (2) die Rate linear zu senken, wenn ein bevorstehender Paketverlust erkannt wird.

MSS maximale Segmentgröße ist die maximale Menge von Anwendungsdaten im Segment (übliche Werte sind 512, 536 und 1.500 Byte).

SampleRTT für ein Segment ist die Zeit zwischen dem Senden des Segments (d.h. Wiedergabe an IP) un dem Empfang einer Bestätigung für das Segment. SampleRTT-

Werte schwanken je nach Überlast in den Routern und verschiedenen Lasten in den Endsystemen von einem Segment zum anderen. Aufgrund dieser Schwankungen TCP verwendet eine Durchnittszeit **EstimatedRTT** der SampleRTT-Werte

$$\text{EstimatedRTT} = (1-x)\cdot \text{EstimatedRTT} + x\cdot \text{SampleRTT}$$
$$\text{Typischer Wert für } x = \tfrac{1}{8} = 2^{-3}$$

EstimatedRTT stellt einen gewichteten Durchschnitt der SampleRTT-Werte dar. Dieser Durchschnitt legt mehr Gewicht auf neuere statt ältere Muster. Der Durchschnitt neuere Mustern wird **Exponetial Weighted Moving Average** EVMA bezeichnet.

Das **Timeout** sollte so gesetzt werden, dass ein Timer nur in seltenen Fällen früh vor der verzögerten Ankunft der Bestätigung eines Segments abläuft. Naturgemäß setzt man das:

$$\text{Timeout} = \text{EstimatedRTT} + 4 \cdot \text{Abweichung (eine gewisse Toleranzspielraum)}$$

Der Spielraum sollte ausreichend groß sein, wenn mit starker Fluktuation in den SampleRTT-Werten zu rechnen ist. Die Auswahl von Faktor 4 (und $\frac{1}{8}$ bei der EWMA ist mehr oder weniger arbiträr, hat aber zwei Vorteile:
1. Multiplikationen um 4 (bzw 2^{-3} bei EWMA) ohne Verschiebung sind möglich
2. Die unnötige Timeouts und Neuübertragung werden vermieden, weil weniger als 1% aller Pakete um mehr als vier Standardabweichungen zu spät ankommen.

Die Abweichung ist eine Schätzung dessen SampleRTT-Werte normalerweise von EstimatedRTT abweicht:

$$\text{Abweichung} = (1-x)\cdot \text{Abweichung} + x \cdot \mid \text{SampleRTT} - \text{EstimatedRTT} \mid$$

Wenn die SampleRTT-Werte wenig Fluktuation aufweisen, dann ist Abweichung klein und Timeout kaum größer als EstimatedRTT. Ist die Fluktuation dagegen groß, ist Abweichung groß und timeout viel größer als EstimatedRTT.

• Das **TCP-Segment** besteht aus **Header-Feldern** und einem **Datenfeld**. Ein TCP-Segment-Header beinhaltet außerdem folgende Felder:

- Portnummer Quelle, Portnummer Ziel.
- Die zwei wichtigsten Felder **Sequenz-** und die **Bestätigungsnummer** sind ein wichtiger Teil des zuverlässigen Datentransfer von TCP.
  Die **Sequenznummer** eines Segments ist die Bytestromnummer des ersten Bytes im Segment.
  Die **Bestätigungsnummer**, die Host A in sein Segment einfügt, ist die Sequenznummer des nächsten Bytes, das Host A von Host B erwartet.
- 16-Bit **Fenstergröße**, 4-Bit-Feld **Header-Länge**, Optionale Feld **Optionen**
- **Flag-Feld** enthält 6 Bit. Die Bits **RST**, **SYN**, und **FIN** werden für den Aufbau und Abbau der Verbindung benutzt. Das **ACK**-Bit spezifiziert, dass der im Bestätigungsfeld enthaltene Wert gültig ist. Ist das **PSH**-Bit gesetzt, weiß der

Empfänger, dass der Daten sofort an die höhere Schicht weiter erreichen soll. Das **URG**-Bit setzt die Priorität für dringende Daten.

TCP konvertiert den unzuverlässigen Dienst von IP zwischen Endsystemen in einen zuverlässigen Datentransportdienst zw. Prozessen.

Für den zuverlässigen Datentransfer verwaltet TCP in den Endsystemen einen Verbindungszustand. Dies beinhaltet Empfangs- und Sendepuffer, Parameter für die Überlastkontrolle sowie Sequenz- und Bestätigungsnummern.

• **Latenz** ist die Zeit von der Einleitung einer TCP-Verbindung durch den Client bis zum Empfang des gesamten angeforderten Objekts beim Client.
− Bei der Übertragung einer kleinen Datei können der Verbindungsaufbau und der Slow-Start von TCP eine beträchtliche Auswirkung auf die Latenz haben.

• **Statisches Fenster:**
Der Server fährt mit dem Senden eines Segment pro Bestätigung fort, bis alle Segmente des Objekts gesendet wurden. Hier sind zwei Fälle zu berücksichtigen:

Fall 1. $W = 4$*:* $\frac{WS}{R} > RTT + \frac{S}{R}$: In diesem Fall empfängt der Server eine Bestätigung für das erste Segment im ersten Fenster, bevor er die Übertragung des ersten Fensters abgeschlossen hat.

$$\textbf{Latenz} = 2 \cdot RTT + \frac{O}{R} \qquad \textbf{minimale Latenz}$$

Fall 2. $W = 2$*:* $\frac{WS}{R} < RTT + \frac{S}{R}$: In diesem Fall überträgt der Server die Segmentmenge des ersten Fensters, bevor er eine Bestätigung für das erste Segment im Fenster empfängt.

$$\textbf{Latenz} = 2 \cdot RTT + \frac{O}{R} + \underbrace{(K-1) \cdot [RTT + (1-W) \cdot \frac{S}{R}]}_{\textbf{Wartezeit des Server}}$$

Gesamt: $\textbf{Latenz} = 2 \cdot RTT + \frac{O}{R} + (K-1) \cdot [\frac{S}{R} + RTT - \frac{WS}{R}]^+$

Wobei $[x]^+ = max(x, 0)$, W ist eine positive Ganzzahl, die ein statisches Überlastfenster mit fester Größe bezeichnet; K sei die Anzahl der Fenster mit Daten, die im Objekt mit Größe O enthalten sind; $\frac{O}{S}$ ist die Anzahl der Segmente im Objekt; Q ist die Zeit, in der der Server wartet, wenn das Objekt eine unendliche Anzahl von Segmenten enthält.

• **Dynamisches Fenster:** TCP nutzt ein dynamisches Fenster

$$\textbf{Latenz} = 2 \cdot RTT + \frac{O}{R} + P \cdot [RTT + \frac{S}{R}] - (2^P - 1) \cdot \frac{S}{R} \textbf{ mit } P = min\{Q, K-1\}$$

▷ Zusammenfassend kann man sagen, dass der Slow-Start die Latenz erheblich erhöhen kann, wenn die Objektgröße relativ klein und dir RTT relativ groß ist.

Verbindungsloser Dienst: UDP

UDP ist verbindungslos, weil es Daten sendet, ohne eine Verbindung aufzubauen.

- UDP ist ein kompaktes Transportprotokoll mit einem minimalistischen Dienst-

modell

- kein Handshake ⇒ keine Verzögerung für den Aufbau einer Verbindung ein.
- unzuverlässiger Datentransferdienst (keine Zusicherungen)
- kein Überlastkontrollmechanismus d.h. der Verkehr wird nicht reguliert.
  Eine Anwendung, die UDP-Transport nutzt, kann in jeder beliebigen Rate senden solange sie will.

Die Segmentstruktur von UDP besteht aus einem Portnummer Quelle , Portnummer Ziel, Länge, Prüfsumme und Anwendungsdaten

Obwohl UDP Fehlerprüfung bietet, unternimmt es nichts, um den Fehler zu beheben. Einige Implementierungen von UDP verwerfen das beschädigte Segment einfach, während andere es mit einer Warnung an die Anwendung weitergeben.

Unterschiede und gemeinsame Merkmale zwischen TCP und UDP

- TCP ist komplexer als UDP
- Beide verwenden Multiplexer und Demultiplexer
- Weder TCP noch UDP bieten keine Zeitzusicherungen.
- TCP verwaltet Verbindungszustand und UDP nicht. Dadurch kann UDP viel mehr aktive Clients unterstützen.
- UDP hat geringere Overhead durch Packet-Header: 8 Byte in UDP und 20 Byte in TCP.
- TCP hat eine regulierte Senderate und UDP nicht
- Eine TCP-Verbindung arbeitet immer Punkt-zu-Punkt, d.h. zwischen einem einzigen Sender und einem einzigen Empfänger. und UDP auch für Multicasting.
- TCP sichert kein min. Übertragungsrate und bietet keinerlei Verzögerungszusicherungen.
- UDP wird für Echtzeit-Anwendungen, Multicasting-Anwendungen, DNS, Routing-Protokolle verwendet und TCP wird für Web, E-Mail Remote Login verwendet.

Jeder Anwendungstyp, der auf einem Endsystem läuft hat eine eindeutige Portnummer. Die Portnummer ist eine 16-Bit-Nummer von $0 - 65535$!

4 Vermittlungsschicht

Die Aufgabe der Vermittlungsschicht ist einfach Pakete von einem sendenden zu einem empfangenden Host transportieren. In diesem Zusammenhang übernimmt die Vermittlungsschicht drei Wichtige Funktionen:

1. **Pfadermittlung:** Die Vermittlungsschicht muss die Route bzw. den Pfad ermitteln, den Pakete von einem Sender zu einem Empfänger nehmen.
2. **Vermittlung:** Wenn ein Paket am Eingang eines Routers ankommt, muss der Router es zur entsprechenden Ausgangsleitung bewegen.
3. **Call-Setup:** Ähnlich wie bei TCP-Verbindung, dass eine Drei-Weg-Handshake geschieht, gibt es auch auf der Vermittlungsschicht (bei Routern) die **Call-Setup**

In einem datagrammorientierten Netzwerk setzt sich die Vermittlungsschicht aus drei wichtigen Komponenten zusammen:

- **Pfadbestimmungskomponente** Routing-Algorithmen
- **Netzwerkprotokoll:** IP-Protokoll
- **Tool:** für die Meldung von Fehlern in Datagrammen (ICMP) und die Beantworten von Anfragen nach bestimmten Informationen der Vermittlungsschicht.

Routing-Algorithmen

Im Kern eines jeden Routing-Protokolls liegt der Routing-Algorithmus, der einen guten Pfad mit geringsten Kosten von der Quelle zum Ziel findet.

Die Knoten des Graphen stellen Router dar, und die Kanten stellen die physikalische Verbindungsleitungen zwischen diese Routern.

Routing-Algorithmen lassen sich allgemein in zwei Typen klassifizieren:

- **Statisch:**
  – Die Routen ändern sich sehr langsam im Verlauf der Zeit.
- **Dynamisch:**
  – Die Routing-Pfade werden entsprechend der Netzwerkverkehrslast oder Änderungen der Netzwerktopologie ändern
  – Reagiert entweder **periodisch** oder als **direkte Reaktion** auf Änderungen der Topologie- oder Verbindungsleitungskosten
  – sind schneller aber auch mit Problemen wie Routing-Schleifen und Routen-Schwankungen

Eine weitere Klassifizierung der Routing-Algorithmen basiert auf globale und dezentrale Arbeitsweise.

- **Link-State-Routing:** (Dijkstra-Algorithmus)
  – ist ein **dynamischer globaler Routing-Algorithmus**.
  – Bevor die Berechnung durchgeführt wird, werden vollständige Informationen über die Kosten der Netzwerktopologie und aller Verbindungsleitungen im voraus verlangt.
  – Der Link-State Algorithmus besteht aus einem Initialisierungsschritt gefolgt von eine Schleife.
  – Schwierigkeiten beim Pathologischen Fall!
  – Die Implementierung von LS hat die Komplexität $\mathcal{O}(n^2)$, der $\mathcal{O}(n \cdot E)$ Nachrichten erfordert.

- **Distanzvektor-Routing** (Bellman-Ford-Algorithmus)
  – **Verteilt:** Jeder Knoten erhält bestimmte Informationen von einem oder mehreren seiner direkt verbundenen Nachbarn, eine Berechnung durchführt und das Ergebnis dann an seine Nachbarn zurück verteilen kann
  – **Iterativ:** Der Prozess setzt so lange fort, bis zwischen den Nachbarn keine Informationen mehr ausgetauscht werden.
  – **Asynchron** Er setzt nicht voraus, dass alle Knoten im Sperrschritt zueinander operieren.
  – **Dynamisch** und **Dezentral**
  – Kein Knoten verfügt über vollständige Informationen über die Kosten alle Netzwerkverbindungen. Stattdessen beginnt jeder Knoten nur mit der Kenntnis der Kosten seiner eigenen, direkt angeschlossenen Verbindungsleitungen.
  – Der DV-Routing-Algorithmus besteht aus einem Initialisierungsschritt und eine Loop-Schleife
  – Distanztabelleneintrag: $D^X(Y, Z) = cost(X, Z) + min_w(D^Z(Y, w))$
  – leidet an dem **Count-to-Infinity-Problem:** Gute Nachrichten werden schnell verarbeitet und die schlechte langsam. Auch mit **Poisoned Reverse** wird das Problem nicht gelöst.
  – DV-Algorithmus führt zur richtigen Lösung schafft aber langsam konvergieren.

- **Weitere Routing-Algorithmen:**
  – **Hot-Potato-Routing:** Der Router versucht, ein abgehendes Paket so schnell wie möglich loszuwerden.
  – **Shortest-Path-First-Routing** ähnlich wie Dijkstra-Algorithmus
  – **Least-Loaded-Path-Routing**
  – **Maximum-Free-Circuit-Routing** ist die Anzahl der freien Kanäle auf einem Pfad das Min. der Anzahl der freien Kanäle jeder Verbindungsleitung des Pfades.

Die wichtigste Unterscheidung zwischen **Routing-Protokollen** im Internet basiert darauf, ob sie für das Routing von Datagrammen innerhalb eines AS oder zwischen mehreren AS benutzt werden.

Router innerhalb des gleichen Autonome Systemen (AS) führen alle den gleichen Routing-Algorithmus. Der Routing-Algorithmus, der innerhalb eines autonomen Systems läuft,

wird als **Intra-AS-Routing-Protokoll** bezeichnet.

Router in einem AS, die für das Routing von Paketen außerhalb des AS zuständig sind, werden als **Gateway-Router** bezeichnet.

Für das Routing innerhalb eines AS im Internet werden drei Routing-Protokolle:

- **RIP** ist ein Distanzvektor-Protokoll
  - RIP ist ein Protokoll der Anwendungsschicht, das auf UDP aufsetzt.
  - In RIP werden Routing-Tabellen zwischen Nachbarn etwa alle 30 sec mit Hilfe einer RIP-Antwortnachricht ausgetauscht.
  - Das Routing-Tabelle besteht aus drei Spalten:
    1. Zielnetzwerk;
    2. Identität des nächsten Routers auf dem kürzestem Pfad zum Zielnetzwerk
    3. Die Anzahl von Hops

- **OSPF** ist der Nachfolger von RIP
  - OSPF ist allerdings ein Link-State-Protokoll
  - Verbesserungen gegenüber RIP:
    1. Sicherheit: alle Informationen werden authentifiziert.
    2. Wenn mehrere Pfade zu einem Ziel die gleichen Kosten aufweisen, lässt OSPF die Verwendung mehrere Pfade zu.
    3. Integrierte Unterstützung von Unicast und Multicast Routing
    4. Hierarchie (4-Typen) innerhalb einer Routing-Domain.

- **EIGRP** ist ein properitärer Routing-Algorithmus von Cisco
  - ist ein Distanzvektor Protokoll

Der Routing-Algorithmus, den Gateways für das Routing zwischen verschiedenen AS verwenden, wird als **Inter-AS-Routing-Protokoll** bezeichnet. Für das Routing zwischen AS im Internet sind paar Versionen von BGP-Routing-Protokolle:

- **BGP Version 4:** ist ein Pfadvektor-Protokoll
  - Die Router beinhaltet kein Kosteninformationen
  - ist ein Standard für Intra-AS-Routing im öffentlichen Internet
  - BGP4 definiert 4 Nachrichtentypen:
    OPEN, UPDATE, KEEPALIVE, NOTIFICATION
  - BGP4 kann auch im Innern eines AS als **Pipe** für den Austausch von BGP-Aktualisierungen zwischen Gateway-Routern, die zum gleichen AS gehören, benutzt werden. Es wird auch Internel-BGP genannt.

IP-Protokoll:

Das IP erledigt drei Hauptaufgaben: Wegwahl, Fragmentierung beim Übergang in Net-

ze mit kleinerer Paketgröße und Flußsteuerung (mit TTL).

Das IP-Protokoll der Vermittlungsschicht bietet eine logische Kommunikation zw. Hosts.

Das IP überträgt Segmente zwischen kommunizierenden Hosts nach "Bestem Bemühen".

Im IP-Dienst können Datagramme Router-Puffer zum Überlauf bringen und nie ihr Ziel erreichen.

IP bietet einen unzuverlässigen Dienst.

- **IP Version 4:**
  ▷ Jede IP-Adresse (32-Bit lang) identifiziert eindeutig das Endsystem.
  ⊖ Insgesamt gibt es 2^{32} mögliche IP-Adressen
  ▷ Die Adressen können nicht beliebig gewählt werden.
  ▷ Es gibt 4 Adressklassen:
  1. **Klasse A:** bis zu 2^7 Netzwerke mit jeweils bis zu 2^{24} Schnittstellen.
  2. **Klasse B:** bis zu 2^{14} Netzwerke mit jeweils bis zu 2^{16} Schnittstellen.
  3. **Klasse C:** verwendet 24 Bit für die Identifizierung des Netzwerks (254 Hosts) und lässt nur 8-Bit für den Schnittstellen Identifizierung.
  4. **Klasse D:** sind für die Multicasting-Adressen reserviert.

  ▷ Lösen des IP-Problem von der CIDR im Jahre 1993 mit a.b.c.d/x wobei x ein Netzwerkpräfix ist. Somit können IP-Adressen beliebig gewählt werden!
  ▷ Jedes IP-Datagramm besteht aus 14 Felder:
  1. **Versionsnummer** möglich 4 oder 6,
  2. **Quelladresse**, **3. Zieladresse**,
  4. **Datenfeld** Reserviert für TCP oder UDP und zusätzlich für ICMP,
  5. **Type-of-Service TOS** um unterschiedliche Typen von IP-Datagrammen voneinander zu unterscheiden,
  6. **Time-to-Live TTL** Dieses Feld soll sicherstellen, das Datagramme nicht für immer und ewig im Netzwerk kreisen,
  7. **Headerlänge**, **8. Datagrammlänge**, **9. Header-Prüfsumme**,
  10. **Optionen** erlauben die Erweiterung eines IP-Header. (selten benutzt)

- **IP Version 6:** "IP new generation"
  ⊕ Kompatibel mit der IP Version 4
  ⊕ Erweiterte Adresierungsmöglichkeiten von 32 auf 128 Bit
  ⊕ Quell- und Zieladresse von 32 auf 128 Bit.
  ⊕ *40-Byte-Header:* schneller Verarbeitung von IP-Datagrammen
  ⊕ Flusskenzeichung und Priorität.
  ⊕ neuer Version von ICMP-Protokoll
  ⊕ Fragmentierung und Reassemblierung sind nicht vorhanden (Zeitaufwendig)
  ⊕ Auch die Prüfsumme ist nicht vorhanden, da in TCP, UDP, Ethernet auch eine Prüfsumme gibt

Problem: Möglicherweise jede Verbindungsleitung auf der Route zwischen Sender und Empfänger verwendet unterschiedliche Sicherungsschichtprotokolle und jeder die-

ser Protokolle eine andere MTU hat. ***Nachteil:*** Ein Last für die Internet-Router:

• **Fragmentierung** kleine Datagramme, die mit eine Identifizierungsnummer sowie ein Quell- und Zieladresse gestempelt werden. TCP als auch UDP erwarten komplette, unfragmentierte Segmente von der Vermittlungsschicht
• **Reassemblierung** Wiederherstellung von kleine Datagramme in kompletten Datagramm.

Übergang von IPv4 auf IPv6

1. Möglichkeit: einen Stichtag auszurufen.

2. Möglichkeit: **Dual-Stack-Ansatz** in dem jeder Knoten IPV6 auch IPv4 Implementierungen aufweist. (siehe Beispiel im Buch)
   Problem Es werden nicht alle ursprüngliche Felder an Ziel ankommen.

3. Möglichkeit: **Tunneling** Beim Tunneling fügt der IPv6-Knoten auf der sendenden Seite des Tunnels das ganze IPv6-Datagramm in das Nutzdatenfeld eines IPv4-Datagramms ein. Dieses IPv4-Datagramm wird dann an den IPv6-Knoten auf der empfangenden Seite des Tunnels adressiert und an den ersten Knoten im Tunnel gesendet.

4. Zusammengefasst sind die Änderungen der Protokolle in der Vermittlungsschicht enorm schwierig.

ICMP:

ICMP wird von Hosts, Routern und Gateways benutzt, um Informationen über unerwartete Situationen auf der Vermittlungsschicht zu übermitteln.

• ICMP wird in der Regel für Fehlermeldungen verwendet.

• ICMP wird oft als ein Teil der IP betrachtet. Ist aber oberhalb der IP angesiedelt.

• ICMP-Meldungen werden genau wie TCP- oder UDP-Segmente als IP-Nutzdaten übertragen.

• ICMP-Meldungen bestehen aus einem TYP- und einem Codefeld

• *Typischer Beispiel:* Ping-Programm sendet ICMP-Meldungen.

Die innere eines Router:

Ein Router besteht aus:

- **Eingangsports:** Der Eingangsport übernimmt mehrere Funktionen der Bitübertragungsschicht, Sicherungsschicht und die wichtigsten Funktionen für Switching-Fabric nämlich Such- und Weiterleitungsfunktion.

- **Switching-Fabric** oder Schaltnetzwerk verbindet Eingangsports mit den Ausgangsports eines Routers. Das Switching lässt sich auf 3 Arten durchführen:

1. Switching über Speicher: einfachste und die älteste Methode. Viele moderne Router vermitteln noch über Speicher.
2. Switching über einem Bus: Bei diesem Ansatz transferieren die Eingangsports ein Paket ohne Intervention des Routing-Prozessors über einen gemeinsamen Bus direkt an den Ausgangsport. Es kann nur ein Paket über den Bus übertragen werden, weil er gemeinsam genutzt wird. Daher enstehen Warteschlangen, und die Busgeschwindigkeit (bis zu Gigabit/sec) ist begrenzt.
3. Switching über Schaltnetzwerk: Um die Bandbreitenbegrenzung eines einzigen, gemeinsam genutzten Bus zu überwinden, bietet die Verwendung eines verbesserten Schaltnetzwerk (Crossbar-Switch). (bis zu 60 Gigabit/sec)

HOL-Blockierung ist ein Phänomen bei einem Switch, dass ein in einer Eingangswarteschlange stehendes Paket muss auf den Transfer durch die Switching-Fabric warten (obwohl sein Ausgangsport frei ist), weil es durch ein anderes Paket am Anfang der Schlange blockiert wird.

- **Routing-Prozessor** führt die Routing-Protokolle, pflegt die Routing-Tabellen und führt Netzwerkmannagementfunktion. Eine Kopie der Routing-Tabelle wird in jedem Eingangsport gespeichert. (→ Vermiedet Weiterleitungsflachenhals)
  − Eine schnelle Suche in einer große Routing-Tabelle ist sehr wichtig:
  Mit Lineare Suche ist unmöglich mit Binäre-Suche sehr langsam 2^{32} und mit CAM in konstanter Zeit mit $log(N)$ Schritte. Der Vorteil dieser Suche liegt, dass die Tabelleneinträge in einem Cache vorgehalten werden. Wobei die Größe des Caches ein Problem darstellt.

- **Ausgangsports:** Ein Ausgangsport speichert die Pakete und Anschließend überträgt er die Pakete an die abgehende Verbindungsleitung. Der Ausgangsport führt die umgekehrte Funktionalität wie der Eingangsport aus

In beiden Ports (Eingang- und Ausgangsport) kommt es zu einem Paketverlust. Die Abarbeitungsstrategien die im Warteschlange verwendet werden sind:

- **FCFS:** Bei einer Warteschlange entscheidet das *Packet-Discarding-Policy*, ob das Paket verworfen wird oder andere Pakete aus der Warteschlange entfernt werden, um für das ankommende Paket Platz zu schaffen.

- **Priority Queuing** Bei der Prioritäts-Warteschlangendisziplin werden Pakete, die an der Ausgangsleitung ankommen, nach einer von zwei oder mehr Prioritätsklassen an der Ausgangswarteschlange klassifiziert. (TOS-Feld)

- **Round-Robin-Queuing** ist ein Work-conserving Queuing der nie zulässt, dass die Verbindungsleitung untätig ist, solange Pakete (irgendeiner Klasse) in einer Warteschlange auf die Übertragung warten.
  Endet die Bedienung eines Pakets nicht innerhalb von t Zeiteinheiten, merkt sich der Server den aktuellen Bedienzeit des Pakets und schiebt ihn in die Warteschlange zurück. Der nächste Paket wählt er nach FIFO-Regel.

- **Weighted Fair Queuing** (WFQ): ist eine generaliserte Abstraktion des Round-Robin-Queuing der mit Prioritäten arbeitet. Router haben für jede Ausgangsleitung mehrere Warteschlangen. WFQ spielt eine besondere Rolle in QoS-Zusicherungen. WFQ unterschiedet sich vom Round-Robin-Queuing dahingehend, dass jede Klasse einen differentiellen Dienstumfang in einem bestimmten Zeitintervall erhalten kann. Das bedeutet, dass jeder Klasse i ein Gewicht w_i zugewiesen wird. Bei einer Verbindungsleitung mit Übertragungsrate R erreicht Klasse i also immer einen Durchsatz von mindestens $R \cdot \frac{w_i}{\sum w_j}$

Router haben keine Ahnung von TCP-Verbindung; sie sehen Datagramme, aber keine Verbindungen.

Multicast

Die Daten von einem Knoten zu einem Host werden in drei Arten gesendet:
Unicast, Broadcast und Multicast

Protokolle, die nur zwischen einem Sender und einem Empfänger operieren, werden als **Unicast-Protokolle** bezeichnet.

Mit Multicast wird Seiten des Servers Ressourcen gespart und auf Seite Netzwerks Bandbreite gespart.

Das Internet-Multicast ist im Gegensatz zum Unicast kein verbindungsloser Dienst. Die Zustandsinformationen für ein Multicast-Verbindung müssen in Router, die Multicast Pakete behandeln, eingerichtet und gepflegt werden. Dies setzt wiederum eine Kombination aus Signalisierungs- und Routing-Protokolle voraus, damit der Verbindungszustand in den Routern aufgebaut, gepflegt und abgebaut werden kann.

Es gibt zwei Ansätze für die Implementierung der Multicast-Verbindungen:
1. Durch Replizierung der Datagramme im Netzwerk-Router
2. Der Sender benutzt eine getrennte Unicast-Transportverbindung zu jedem Empfänger.

Problem einer Multicast-Kommunikation ist, wie man die Empfänger eines Multicast-Datagramms identifiziert und wie muss ein Datagramm an diese Empfänger adressiert werden.

Multicast auf der Vermittlungsschicht im Internet setzt sich aus zwei Komponenten zusammen:

1. IGMP-Protokoll Version 2 operiert an der Netzwerkperipherie zwischen einem Host und einem direkt angeschlossenen Router. IGMP stellt einem Host die Mittel zur Verfügung, damit dieser seinen angeschlossenen Router darüber informieren kann, dass eine auf ihm laufende Anwendung einer bestimmten Multicast-Gruppe beitreten möchte.

Wie ICMP- werden auch IGMP-Nachrichten in einem IP-Datagramm verkapselt und erhalten IP-Protokollnummer

Das Ziel von Multicast-Routing ist es dann, einen Baum von Verbindungsleitungen

zu finden, die alle Router mit angeschlossenen Hosts verbinden, die Mitglieder der Multicast-Gruppe sind. In der Praxis wurden zwei Ansätze für die Ermittlung des Multicast-Routing-Baums übernommen. Die beiden Ansätze unterscheiden sich danach, ob ein einzelner Baum benutzt wird, um den Verkehr für alle Sender der Gruppe zu verteilen, oder ob ein quellenspezifischer Routing-Baum für jeden einzelnen Sender gebildet wird:
• ***Gemeinsamer Gruppenbaum:*** Bei diesem Ansatz wird nur ein Routing-Baum für die gesamte Multicast-Gruppe gebildet.
Die Ermittlung eines Baumes mit minimalen Kosten wird als *Steiner-Baum-Problem* bezeichnet (NP-vollständig). Obwohl für das Steiner-Baum-Problem gute Heuristiken vorliegen, gründet interesanterweise keiner der im internet existierenden Multicast-Routing-Algorithmen auf diesem Ansatz. Grund ist, dass Informationen über alle Verbindungsleitungen im Netzwerk erforderlich ist. Ein weiterer Grund ist, dass Algorithmus bei jeder Änderung der Verbindungsleitungskosten erneut ausgeführt werden muss, um den Baum mit minimalen Kosten quasi aufzufrischen.
Bei der alternativen *zentrumbasierten Ansatz* wird ein Zentrumsknoten für die Multicast-Gruppe gesucht. Ein Baum mit den geringsten Pfadkosten ist nicht gleich dem Baum mit den minimalen Gesamtkosten!
• ***Quellenbasierte Bäume:*** Bei diesem Ansatz wird für jeden Sender der Multicast-Gruppe ein getrennter Routing-Baum gebildet. In einer Multicast-Gruppe mit N Hosts werden also N verschiedene Routing-Bäume für eine Multicast-Gruppe gebildet.

Der Multicast-Routing-Algorithmus mit den geringsten Pfadkosten ist ein Link-State-Algorithmus. Er setzt voraus, dass jeder Router den Zustand jeder Verbindungsleitung im Netzwerk kennt, um den Baum mit den geringsten Pfadkosten von der Quelle zu allen Zielen berechnen zu können.
Ein einfacher Multicast-Algorithmus, der weniger Zustandsinfromationen als der Routing-Algorithmus für die geringsten Pfaden erfordert, ist der **R**everse-**P**ath-**F**orwarding-Algorithmus. Dieser Algorithmus setzt nicht voraus, dass ein Router den vollständigen kürzesten Pfad von sich zur Quelle kennt.
Die Lösung für das Problem des Empfangs unerwünschter Multicast-Pakete unter RPF wird als *Pruning* bezeichnet. Pruning setzt voraus, dass ein Router weiß, welche Router in Downstream-Richtung in Bezug auf den Empfang ihrer Multicast-Pakete von ihm abhängen.

2. Multicast-Routing-Protokoll

- **DVMRP:** Ist ein Distanzvektor-Algorithmus, der es jedem Router ermöglicht, die Ausgangsverbindungsleitung zu berechnen, die den kürzesten Pfad zurück zu jeder möglichen Quelle darstellt. DVMRP hat sich als Protokoll für Inter-AS-Multicast-Routing durchgesetzt.

- **PIM** ist protokollunabhängig
  verwendet zwei unterschiedliche Multicast-Verteilungsszenarien:

1. Dense-Mode, in dem Gruppenmitglieder dicht beieinander befinden. Dieser Modus ist eine RPF-Technik mit Fluten und **Pruning** und ist ähnlich mit DVMRP
2. Im **Sparse-Mode** ist die Anzahl der Router mit angeschlossenen Gruppenmitgliedern im Vergleich zur Gesamtzahl der Router gering und die Gruppenmitglieder sind weitflächig. Dieser Modus ist ein zentrumbasierter Ansatz.

- **MOSFP** wird in AS in dem OSPF-Protokoll benutzt wird, für das Unicast-Routing eingesetzt.
- **CBT** ist ein Multicast-Routing Protokoll, das einen bidirektionalen gemeinsamen Gruppenbaum mit einzigem Kern (Zentrum) konstruiert.

Die Grenze zwischen dem Host und der physikalischen Verbindungsleitung nennt man **Schnittstelle**

5 Sicherungsschicht

Die Sicherungsschicht ist dafür zuständig, das ein Datagramm über eine einzelne Verbindungsleitung zu transferieren.

Die Sicherungsprotokolle werden in größtenteils in Adaptern oder in Netzwerkkarten implementiert. Ein **Adapter** besteht aus:

- **Busschnittstelle:** Zuständig für die Kommunikation mit dem Elternknoten des Adapters
- **Leitungsschnittstelle:** Verantwortlich für die Implementierung des Sicherungsschichtprotokolls

Obwohl es mehrere Sicherungsschichtprotokolle gibt, können sie miteinander kommunizieren. Ein Sicherungsschichtprotokoll übernimmt beim Senden und Empfangen von Rahmen:

- Fehlererkennung und Fehlerkorrektur (siehe Abbildungen und Formeln im Buch!)
  Es gibt drei Techniken zur Erkennung von Fehlern:
  1. Daten-Paritätsprüfungen: einfachste Form; gerader Parität und ungerader Parität.
  2. Prüfsummenmethoden: die vorwiegend auf der Transportschicht angewendet werden. Die Datenbytes werden als 16-Bit-Ganzzahlen behandelt und ihre Einerkomplementsumme bildet die Internet-Prüfsumme.
  3. Zyklische Redundanzprüfungen: die vorwiegend auf der Sicherungsschicht angewendet werden

- Flusskontrolle und Zufallszugriff
- Zuverlässige Übertragung ähnlich wie auf der Transportschicht mit Neuübertragung und Bestätigungen. Ein Protokoll der Sicherungsschicht kann eine zuverlässige Übertragung bieten und ein anderes nicht.
- Halb- und Vollduplex-Übertragung

Geräte der Bitübertragungsschicht

- Ein **Repeater** ist ein Gerät der Bitübertragungsschicht, das mit einzelnen Bits statt Rahmen arbeitet. Ein Repeater hat zwei oder mehr Schnittstellen. Wenn ein Bit, das eine Null oder Eins darstellt, von einer Schnittstelle ankommt, reproduziert der Repeater das Bit, womit er seine Energiestärke erhöht, und überträgt es an alle anderen Schnittstellen.
  ▷ Repeater werden häufig in LANs eingesetzt.
  ▷ Beim Einsatz der Repeatern in einem Ethernet, implementieren sie kein CS oder CSMA/CD.
- **Hubs** sind einfache Geräte, um Netzwerke zu verbinden.
  Hubs sind im Wesentlichen Repeater, die mit Bits arbeiten. Wenn ein Bit an einer Hub-Schnittstelle ankommt, sendet der Hub das Bit einfach rundum an alle anderen Schnittstellen. Hubs haben aber keinen Einfluss auf die Verfügbare Bandbreite im Netzwerk.
  ▷ **Backbone-Hubs** vergrößert max. Entfernung.

Geräte der Sicherungsschicht, die LAN's verbinden

- **Bridges** arbeiten mit Ethernet-Rahmen und leiten sie weiter zur Sicherungsschicht.
  ▷ haben wenige Schnittstellen (zwei bis vier)
  ▷ Bridges sind Paket-Switches nach dem Store-and-Forward-Prinzip, die die Rahmen unter Verwendung der LAN-Zieladressen weiterleiten und filtern.
  ▷ **Filterung** stellt fest, ob ein Rahmen an eine Schnittstelle weitergeleitet oder verworfen soll.
  ▷ **Weiterleitung** ist die Fähigkeit, die Schnittstelle zu ermitteln, an die ein Rahmen zu senden ist.
  ▷ Das Filtern und Weiterleiten einer Bridge erfolgt anhand einer **Bridge-Tabelle**. Die Bridge-Tabelle enthält Einträge für einige, aber nicht unbedingt alle Knoten eines LAN. Der Eintrag eines Knoten in der Bride-Tabelle enthält:
  1. LAN-Adresse des Knoten,
  2. Bridge-Schnittstelle, die in Richtung zum Knoten führt und
  3. Zeit, wann der Eintrag für den Knoten in die Tabelle eingefügt wurde.
  ▷ Um das Kreisen und Multiplizieren von Rahmen zu verhindern, d.h um eine

Topologie ohne Schleifen zu ermitteln, wird ein Spanning-Tree-Protokoll verwendet
⊕ Überwinden viele Probleme an denen Hubs leiden.
⊕ erreichen relativ hohe Raten bei der Paketfiltrierung und -weiterleitung
⊙ Theoretisch ist es mit Bridges möglich, ein LAN aufzubauen, das sich über den gesamten Globus erstreckt aber große Netzwerke benötigen auch Router.
⊖ Bridges bieten keinen Schutz vor Broadcast-Fluten

- **Switches**
  ⊕ Sind im Wesentlichen leistungsstarke Bridges mit mehreren Schnittstellen.
  ⊕ Wie bei Bridges, filtern und leiten sie Rahmen anhand von LAN-Zieladressen weiter.
  ⊕ Switches haben ein Dutzende Schnittstellen, die eine hohe Gesamtweiterleitungsrate durch die Switching-Fabric erzeugen.
  ⊕ Viele Switches arbeiten Vollduplex. Das heißt, sie können Rahmen gleichzeitig über die gleiche Schnittstelle senden und empfangen.
  ⊕ Switches verwenden **Cut-Through-Switching** statt Store-and-Forward-Prinzip. Der Unterschied ist sehr gering (0.12 bis 1.2 ms). Der Cut-Through-Switch betrachtet bei jeder ankommenden Nachricht nur die Zieladresse und sendet die Nachricht dann direkt an das entsprechende Zielsegment weiter.
  ⊕ Sind Plug-and-Play Geräte

Es gibt zwei Typen von **Netzerkleitungen**:

- **Punkt-zu-Punkt Leitung (Point-to-Point):** besteht aus ein einzigen Sender und Empfänger, die ausschließlich über diese Leitung kommunizieren.
- **Broadcast Leitungen:** können mehrere sendende und empfangende Knoten über den gleichen gemeinsamen genutzten Broadcast-Kanal verbunden sein.

Mehrfachzugriffsprotokolle

Ein Mehrfachzugriffsprotokoll für einen Broadcast-Kanal mit einer Rate von R Bps im Idealfall weist folgende Merkmale auf:

- Wenn nur ein Knoten Daten zu senden hat, hat dieser Knoten eine Durchsatz von R bps
- Wenn M Knoten Daten zu senden haben, hat jeder dieser Knoten einen Durchsatz von $\frac{R}{M}$ bps
- Das Protokoll ist dezentral, d.h. es gibt kein Ausfall von Master-Knoten
- Das Protokoll ist einfach und die Implementierung ist Kostengünstig.

Es gibt drei allgemeine Ansätze für die Koordination des Zugriffs auf einem Broadcast-Kanal:

1. **Kanallaufteilungsprotokolle:**
   1.1 TDM beseitigt Kollisionen und ist absolut fair. Bei TDM enthält jeder Adapter periodisch in kurzen Zeitintervallen die gesamte Bandbreite
   Nachteile: Erstens ein Knoten wird auf eine Durchschnittsrate von R/N bps eingeschränkt und zweitens ein Knoten muss immer warten, bis er in der Übertragungssequenz an die Reihe kommt, auch wenn er wiederum der einzige Knoten ist der einen Rahmen zu senden hat.
   1.2 FDM enthält jeder Adapter kontinuierlich einen Anteil an der Bandbreite. FDM Weist die gleichen Vor- und Nachteile wie TDM auf.
   1.3 Code-Division-Multiple-Access Unter CDMA wird jedem Knoten ein unterschiedlicher Code zugewiesen. Jedes vom Sender übertragene Bit wird kodiert, dass das Bit mit dem Code multipliziert wird.
   Vorteil: CDMA ermöglicht es mehrere Konten zu übertragen
   Nachteil: Die Bitrate ist sehr niedrig.

2. **Zufallszugriffsprotokolle:**
   Bei einem Zufallszugriffsprotokoll überträgt ein sendender Knoten in der vollen Kanalrate, d.h R bps. Tritt eine Kollision auf, überträgt jeder von der Kollision betroffene Knoten seinen Rahmen so oft erneut, bis der Rahmen kollisionsfrei durchkommt. Wenn ein Konten aber an einer Kollision leidet überträgt er nicht unbedingt den gleichen Rahmen sofort noch einmal. Vielmehr wartet er eine zufällige Verzögerung, bevor er den Rahmen erneut überträgt.
   2.1 ALOHA Erste ALOHA-Protokoll basiert nicht auf Zeitschlitzen und war voll dezentralisiert.
   $\oplus$ Kein anderer Knoten kann eine Übertragung starten, während Knoten i überträgt, weil sich eine solche Übertragung mit dem letzten Teil der Übertragung von Knoten i überlappen würde. Da beim reinem ALOHA eine Station den Kanal vor dem übertragen nicht abfragt, hat sie keine Möglichkeit zu erfahren, dass ein anderer Rahmen schon unterwegs ist. Die W'keit, dass ein bestimmter Knoten erfolgreich übertragen kann, ist $p \cdot (1-p)^{2\cdot(N-1)}$ und die max. Effizienz $\frac{1}{2\cdot e}$ (die Hälfte des S-ALOHA)
   2.2 Slotted-ALOHA ist ein extrem einfaches Protokoll
   $\oplus$ Beim unterteilen von ALOHA wird die Effizienz verdopelt. die Zeit wird in einzelne Intervalle geteilt, wobei jedes Intervall einen Rahmen entspricht.
   $\ominus$ S-ALOHA setzt voraus, dass alle Konten ihre Übertragung so synchronisieren, dass sie am Anfang eines Schlitzes beginnen.
   $\oplus$ Funktioniert gut, wenn nur ein Knoten aktiv ist.
   $\oplus$ Wenn kein Form der Zugriffskontrolle angewandt und jeder Knoten nach jeder Kollision sofort erneuert übertragen würde, wäre die Effizienz Null.
   $\ominus$ Wenn N Knoten aktiv sind ist die W'keit $N \cdot p \cdot (1-p)^{N-1}$ und
   die max. Effizienz ist $\frac{1}{e} = 0.37$ d.h. Der Kanal kann nur 37 % ausgenutzt
   2.3 CSMA und CSMA/CD
   Bei CSMA/CD werden keine Zeitschlitzen benutzt.
   Carrier Sense: Adapter prüft ob die Leitung frei ist, wenn nicht dann überträgt

er nie einen Rahmen.
Multiple Access: Schickt die Daten, allerdings können noch andere gleichzeitig Daten aufs Netz schicken
Collision Detect: Ein momentan aktiver Knoten bricht seine Übertragung ab, sobald er feststellt, dass ein anderer Adapter ebenfalls überträgt. Falls eine Kollision stattfindet, wird sie erkannt und nach einer zufälligen Zeitspanne werden die Daten noch einmal geschickt.
▷ Mit der **Jam-Signal** soll sichergestellt werden, dass alle anderen übertragenden Adapter von der Kollision erfahren.
▷ Zufallsteuerung im Falle einer Kollision:
Nach der Übertragung des Jam-Signals tritt der Adapter in eine **exponentielle Backoff-Phase**. Das heißt, wenn er einen bestimmten Rahmen überträgt und dieser Rahmen nacheinander auf n Kollisionen gestoßen ist, wählt der Adapter einen Wert für K zufällig aus $\{0, 1, 2, \ldots 2^{m-1}\}$, wobei $m : min(n, 10)$. Der Adapter wartet dann $K \cdot 512$ Bitzeiten und fährt anschließend mit Schritt 2 (Carrier Sense) fort.
⊕ In einer LAN-Umgebung ist CSMA/CD besser als Slotted-ALOHA
⊕ Wenn die max. Ausbreitungsverzögerung zwischen Stationen sehr gering ist, kann die Effizienz von CSMA/CD nahezu 100% betragen.
⊕ Im Gegensatz zu ALOHA können die Nachrichten bei CSMA/CD unterschiedlich lang sein, wobei das Minimum 64 Bytes und das Maximum 1518 Bytes beträgt. *Hinweis* Das Risiko für Übertragungsfehler und die damit verbundene Wiederholung im wesentlichen proportional zur Länge der Nachricht steigt.
⊖ garantiert nicht die maximale Übertragungsrate
⊖ Für Echtzeit-Anwendungen nicht geeignet. (setzt keine Prioritäten für Nachrichten)

3. **Rotationsprotokolle:**
3.1 Polling-Protokoll: Es setzt voraus, dass einer der Knoten als Master-Knoten fungiert. Der Master-Knoten pollt die einzelnen Knoten rundum ab.
⊕ Vermeidet die Kollisionen und die leeren Schlitze
⊖ führt eine Polling-Verzögerung, d.h. die erforderliche Zeit für die Benachrichtigung eines Knoten, dass er übertragen kann.
⊖ Wenn der Master-Knoten ausfällt, bricht der Betrieb des gesamten Kanals zusammen.
3.2 Token-Passing-Protokoll: Token ist ein kleiner spezieller Rahmen, dass zwischen den Knoten in einer festgelegten Reihenfolge ausgetauscht werden.
⊕ Da immer nur ein Station das Token besitzt, treten keine Kollisionen auf.
⊕ Es gibt kein Master-Knoten
⊕ Ist dezentral und sehr effizient
⊖ Der Ausfall eines Knoten kann den gesamten Kanal zum Absturz bringen.
⊖ Wenn ein Knoten versehentlich nicht mehr freigibt, muss eine Wiederherstellungsprozedur aktiviert werden, um das Token wieder in Umlauf zu bringen.

Ethernet: ist ein Broadcast-Leitungstechnologie
▷ 1970 Entwickelt; 1976 öffentlich vorgestellt.
▷ benutzt das CSMA/CD Mehrfachzugriffsprotokoll
▷ Ein Ethernet-LAN kann ein Bus- oder Sterntopologie aufweisen.
⊕ ist viel einfacher und kostengünstiger als Token-Ring, FDDI, und ATM Netzwerke
⊕ Ethernet reagiert immer mit neueren Versionen, die eine höhere Datenrate bieten
⊕ Übertragungsrate: 10 Mbps, 100 Mbps, 1 Gbps und neuste 10 Gbps.
▷ **Effizienz:** Wenn nur ein Knoten einen Rahmen zu senden hat, kann der Knoten in der vollen Rate übertragen.
Müssen mehrere Knoten Rahmen übertragen, dann ist die approximative Effizienz:

$$\text{Ethernet-Effizienz} = \frac{1}{1 + 5 \cdot \frac{t_{prop}}{t_{trans}}}$$

• Wobei t_{prop} die maximale Zeit, bis sich Signalenergie zwischen jeweils zwei Adaptern ausbreitet und t_{trans} die Zeit für die Übertragung eines Ethernet-Rahmens mit maximaler Größe ist.
• Wenn im ersten Fall $t_{prop} \to 0$ (d.h. die Ausbreitungsgeschwindigkeit geht $\to 0$) oder auch im zweiten Fall $t_{trans} \to \infty$ dann nähert sich die Effizienz gegen 1.

▷ Die Ethernet-Effizienz hängt ab von der Kabellänge, Bandbreite, Paketgröße

▷ Alle Ethernet-Technologien haben die gleiche Rahmenstruktur:
Quell- und Zieladresse, Typfeld, Datenfeld, CRC-Feld für Erkennung von Fehlern und Präambel (8 Byte) die ersten 7 Byte dienen zum Aufwecken und letztes Byte zum Warnen.

▷ Ethernet Pakete können nicht mehr als 1.500 Datenbyte umfassen und die Pakete für viele Weiterstreckenleitungen nicht mehr als 576 Byte unterstützen.

Alle Ethernet-Technologien bieten der Vermittlungsschicht einen **verbindungslosen Dienst** (ohne Handshake)

▷ Alle Ethernet-Technologien bieten der Vermittlungsschicht einen **unzuverlässigen Dienst**. (kein Bestätigung für den Empfang)

• Es gibt drei Adresstypen: Hostnamen für die Anwendungsschicht, IP-Adressen für die Vermittlungsschicht und LAN-Adressen für die Sicherungsschicht.

• Eine **LAN-Adresse** wird auch als physikalische, Ethernet- oder MAC-Adresse bezeichnet. Bei den meisten LANs ist eine LAN-Adresse sechs Byte lang, so dass 2^{48} mögliche LAN-Adressen zur Verfügung stehen. Diese 6-Byte-Adressen werden normalerweise in hexadeximaler Notation ausgedrückt. Zwei Adapter haben **nie** eine gleiche LAN-Adresse.

Adressauflösungsprotokoll (ARP)

ARP ist ein Übersetzter zwischen der Adressen der Vermittlungsschicht (IP-Adressen) und der Adressen der Sicherungsschicht (LAN-Adressen) und umgekehrt.

▷ In jedem ARP-Modul befindet sich ein **ARP-Tabelle** und **Time-To-Live**-Eintrag

⊕ Für die Knoten im gleichen LAN löst das ARP den IP auf.
⊕ **Plug-Play-Protokoll**: ARP-Tabelle wird automatisch aufgebaut.

MAC-Schicht und Medienzugriffsprotokolle

- Media-Access-Control: Zugriff von Stationen auf ein Medium.
- MAC-Schicht ist der untere Teil von Sicherungsschicht.
- Beschreibt außerdem drahtlose Netze, die eine niedrige Kapazität (1 Mbps 2 Mbps) als die verdrahten Netze haben.
- **CSMA/CA (Collision Avoidance)** ist ein MAC-Protokoll
  − CSMA/CA findet seinen Einsatz in LANs mit geringer Anzahl von Stationen, die alle bekannt und immer aktiv sind. Hierbei sind die Stationen entsprechend ihrer Priorität von 1 bis n durchnumeriert, und es wird eine Zählersteuerung benutzt.
  Für den korrekten Empfang der Datenrahmen sendet der Empfänger nach eine kurze Zeit eine explizite Bestätigung an dem Sender.
  CSMA/CA implementiert keine Kollisionserkennung. Grunde:
  1. Teuer: Die Fähigkeit, Kollisionen zu erkennen, setzt das gleichzeitige Senden und Empfangen voraus.
  2. Auch wenn man Kollisionserkennungen hätte und beim Senden keine Kollisionen feststellt, kann trotzdem beim Empfänger ein Kollision auftreten.
- **Fading** (Schwund) bedeutet, dass die Stärke des Signals während seiner Ausbreitung durch das drahtlose Medium abschwächt.
- Das **MAC-Zugriffsprotokoll CSMA/CD** kann auch einen kurzen **R**equest-**T**o-**S**end-Steuerrahmen und einen kurzen **C**lear-**T**o-**S**end-Rahmen verwenden, um Zugriff auf den Kanal zu reservieren.
  Die vom Sender gesendete RTS-Rahmen können Daten- und ACK-Pakete enthalten. Der Empfänger antwortet mit einem CTS-Rahmen, in dem er erlaubt Daten zu übertragen und ohne Kollision.

Point-To-Point Protokoll

- PPP-Protokoll ist für die Wählleitung von privaten Hosts an einen Provider bevorzugt z.B. Die Verbindung mit dem Modem, X.25, ISDN
- PPP-Protokoll muss in der Lage sein, mehrere Protokolle der Vermittlungsschicht, mehrere Leitungsarten, zu unterstützen.
- Von PPP wird verlangt, Bitfehler zu erkennen, es muss aber nicht korrigieren.

- Die Wichtigsten Komponente für die Fehlermeldung und Abschaltung einer PPP-Leitung sind:
  1. Framing Methode für die Daten-Verkapselung und Fehlererkennung in einem PPP-Rahmen
  2. LCP Link-Control-Protocol: Für die Initialisierung, Wartung und Abbau der PPP-Verbindungsleitung
  3. NCPs Network-Control-Protocol: Für die selbst Konfigurierung der Module auf der Vermittlungsschicht.

ATM-Netzwerke

- Die ATM-Standards definieren Paketvermittlung mit virtuellen Kanälen.
- Wegen ihrer hohe Datenrate (Terabits pro sec) werden in Telefonnetzen und in Internet-Backbones angewendet und nicht in Desktop-PCs und Workstations
- Im ATM-Bereich werden Pakete mit einer festen Länge von 53 Byte Zellen genannt.
- ATM verwendet eine eigene Protokollhierarchie, das aus drei Schichten besteht:
  1. ATM-Bitübertragungsschicht befasst sich mit Spannung, Bitzeiteingaben und Framing auf dem physikalischen Medium.
  2. ATM-Schicht: Der Kern des ATM-Schicht.
  3. ATM-Adaptionschicht (AAL): entspricht grob der Transportschicht.

X.25 ist die erste öffentliche Paketvermittlungstechnologie. Bei X.25 werden Zustandsinformationen geführt.

Frame-Relay ist der Nachfolger von X.25. Beide WAN-Technologien sind im Aussterben.

Die maximale Datenmenge, die ein Paket der Sicherungsschicht befördern kann, wird als **maximale Transfereinheit MTU** bezeichnet.

6 Multimedia-Vernetzung

Multimedia-Anwendungen reagieren sehr empfindlich auf Ende-zu-Ende Verzögerung und Verzögerungsschwankungen, sie können aber gelegentlich Datenverlust tolerieren. Die Verzögerungen bis 150 ms sind unauffällig; zwischen 150 bis 400 ms sind noch akzeptabel und ab 400 ms sind nicht mehr erlaubt.

Streaming vermeidet, dass zuerst die ganze Datei heruntergeladen muss, bevor man mit der Wiedergabe beginnt.

- **Streaming, gespeichertes Audio und Video:**
  – Der multimediale Inhalt ist im voraus auf einem Server gespeichert.
  – Die Wiedergabe des Audio/Videos beginnt ein paar sec nach Empfang der Datei vom Server.
  – Nachdem die Wiedergabe begonnen hat, sollte sie dem ursprünglichen Rate fortsetzen.
  ▷ Vor der Übertragung werden Dateien Segmentiert und in Header verkapselt.
  ⊕ hohe Qualität
  ▷ Media-Player, Real-Player

- **Streaming von LIVE-Audio und Video** ähnlich mit Radio- und Fernsehübertragung.

- **Interaktives Echtzeitaudio und -video**
  – In Echtzeit zu kommunizieren.
  – Neuübertragungsmechanismen gelten für interaktive Echtzeit Audioanwendungen wie Internet-Phone aber oft als unakzeptabel

Real-Time Protokoll (RTP)
- RTP kann als Schicht der Transportschicht betrachtet werden.
- Aus Sicht der Anwender ist RTP ein Teil der Anwendungsschicht.
- RTP ist ein Protokoll für die Verkapselung von Multimedia-Segmenten
- RTP setzt sich normalerweise auf UDP auf.
- RTP-Pakete sind nicht auf Unicast-Anwendungen begrenzt
- Die vier wichtigen Felder im RTP-Header sind:
  Nutzdaten, Sequenznummer, Zeitstempel und Quellenindentifizierung.
- RTP bietet an sich keine Mechanismus, um die zeitgerechte Zustellung von Daten oder andere Dienstqualitäten zusichern.

⊖ Ton und Bild von Video-Streaming können getrennt gespeichert.

Real-Time Streaming Protocol (RTSP)
- RTSP ermöglicht Vor- und Zurückspulen, Pause usw ähnlich wie bei einem Videorecorder
- RTSP ist ein Out-of-Band Protokoll
- RTSP-Kanal ähnelt auf vielerlei Art dem Steuerkanal von FTP

• RTSP definiert nicht, wie Audio/Video in Paketen für die Übertragung in einem verkapselt werden.
• RTSP bestimmt nicht wie Streaming-Daten transportiert werden; es unterstützt sowohl TCP als auch UDP
• RTSP definiert keine Kompressionsschemata für Audio/Video

Real-Time Control Protocol (RTCP)
• für Multicasting geeignet.
• RTCP ist ein Protokoll, das eine vernetzte Multimedia-Anwendungen in Verbindung mit RTP benutzten kann.
• RTCP verkapseln keine Audio/Videoblöcke
• RTCP-Pakete werden von jedem Teilnehmer einer RTP-Sitzung an alle übrigen Teilnehmer über IP-Multicast übertragen
• RTCP-Pakete werden periodisch gesendet und enthalten Sender und/oder Empfängerberichte in denen Statistiken angekündigt werden, die für die Anwendung nützlich sei können.
• Diese Statistiken beinhalten die Anzahl der gesendete Pakete, die Anzahl der verlorene Pakete und Netzwerk Jitter.

H.323 ist ein Standard für Echtzeit und Videokonferenzen zwischen Endsystemen im Internet.
• H.323-Endpunkte können Web-Telefone, Web-TVs, Internet-.Phone oder Videokonferenzsoftware sein.
• Ein H.323 beinhaltet auch Gateways und Gatekeeper.
• RTP und RTCP ist ein Bestandteil des H.323

Ein **Gatekeeper** ist ein optionales H.323-Gerät. Jeder Gatekeeper ist für eine H.323-Zone zuständig.

Unter Paket-Jitter versteht man die Schwankung von Paket-Verzögerungen innerhalb des gleichen Paketstroms.

Beseitigung von Jitter:
1. Der Sender erhöht die **Sequenznummer** für jedes Paket um eins.
2. Der Sender stempelt jeden Block mit der **Zeit** seiner Erzeugung.
3. Die **Wiedergabeverzögerung** der empfangenen Audioblöcke muss ausreichend lang sein, damit die meisten Pakete vor der geplanten Wiedergabezeit empfangen werden
Feste Wiedergabeverzögerung: Bei dieser versucht der Empfänger, jeden Block genau q ms nach seiner Erzeugung wiederzugeben.
Adaptive Wiedergabeverzögerung: Generischer Algorithmus

Wiederherstellung nach einem Paketverlust

- **Forward Error Correction (FEC)** Die Fähigkeit des Empfängers, Fehler sowohl zu erkennt als auch direkt zu korrigieren, wird als FEC bezeichnet. FEC verringern die Anzahl der Neuübertragungen durch den Sender.
  Es gibt 2 Versionen von FEC:

1. Nach jeweils n Blöcken wird ein redudant kodierter Block gesendet. Der redudanter Block wird durch Ausführung von XOR auf die n Originalblöcke gebildet.
2. Ein Audiostrom wird mit einer geringeren Auflösung als redudanter Information gesendet. z.B. nominelle Audiostrom 64 Kbps und redudante Audiostrom geringere Qualität 13 Kbps.

- **Interleaving:** Durch Verzahnung lässt sich die Wirkung von Paketverlusten mildern. Das Qualität lässt sich beträchtlich verbessern. *Nachteil* der Verzahnung ist ein Erhöhung der Latenz.

Policing ist die Regulierung der Rate, in der ein Fluss Pakete in das Netzwerk einspeisen darf. Es lassen sich drei wichtige Policing-Kriterien identifizieren, die sich je nach Zeitrahmen, über den der Paketfluss reguliert wird, voneinander unterscheiden:
1. Durchschnittsrate: Wünschenswert für das Netzwerk ist die langfristige Durchschnittsrate.
2. Spitzenrate: Die max. Anzahl von Paketen, die über eine kürzere Dauer gesendet werden können
3. Burst-Größe: Unter Umständen kann auch eine Begrenzung der max. Anzahl von Paketen erwünscht sein.

Der **Leaky-Bucket**-Mechanismus ist eine Abstraktion, die zur Charakterisierung dieser Policing-Grenzen verwendet werden kann. Leaky-Bucket besteht aus einem Warteschlangensystem und einem Bucket (Eimer), der bis zu b Token aufnehmen kann. In diesen Bucket werden Token wie folgt eingegeben: Neue Token, die potenziell in den Bucket eingefügt werden können, werden immer in einer Rate von r Token pro Sec erzeugt. Token-Erzeugungsrate r dient für die Begrenzung der langfristigen Durchnittsrate, in der das Paket in das Netzwerk einfließen kann. Ist der Bucket mit weniger als b Token gefüllt, dann wird das neu erzeugt Token zum Bucket hinzugefügt; andernfalls wird es ignoriert und der Token-Bucket bleibt mit b Token gefüllt.
▷ Leaky-Bucket im Verbindung mit WFQ bieten nachweisbare max. Verzögerung in einer Warteschlange.
Wenn b_1 Pakete in der Warteschlange stehen und Pakete in einer Rate von mindestens $R \cdot w_1/(\sum w_j)$ Paketen pro Sekunde aus der Warteschlange bedient werden, dann kann die Zeit, bis das letzte Bit des letzten Pakets übertragen wurde, nicht länger als sein $d_{max} = \frac{b_1}{R \cdot w_1/(\sum w_j)}$

Signalisierungsprotkoll RSVP ermöglicht den auf Hosts laufenden Anwendung, Ressourcen (in diesem Fall nur Bandbreite) im Internet zu reservieren.
• RSVP ist kein Routing-Protokoll!
• RSVP wird auch von den Routern verwendet, um Anfragen für Bandbreitenreservierungen weiterzuleiten.
• Um RSVP zu implementieren, muss RSVP-Software in den Empfängern, Sendern und Routern vorhanden sein.
• RSVP bietet Reservierungen für Bandbreite in Multicast-Bäumen.
• RSVP ist empfangsorientiert, d.h., der Empfänger eines Datenflusses leite die Res-

sourcenreservierung für den betreffenden Fluss ein.
• RSVP-Nachrichten werden in das Informationsfeld des IP-Datagramms eingefügt (Kann verloren gehen)

Integrated-Services dient für die Bereitstellung von Dienstqualität im Internet realisiert zu werden.
⊖ Intserv-Architektur müssen in den die Routern Pro-Fluss-Zustand verwalten.
▷ Intserv-Architektur definiert zwei wichtige Dienstklassen:

- **Zugesicherte Dienstqualität (QoS)** Die zugesicherte Dienstqualität biete feste (mathematisch nachweisbare) Grenzen für die Warteschlangenverzögerung, denen ein Paket in einem Router ausgesetzt. Die Verkehrcharakterisierung einer Quelle wird durch einen Leaky-Bucket mit den Parametern (r, b) angegeben und der angeforderte Dienst wird durch eine Übertragungsrate R Übertragungsrate R charakterisiert, in der Pakete übertragen werden.

- **Controlled-Load-Dienst** Mit CL-Dienst erhält eine Sitzung eine Dienstqualität, die fast der Dienstqualität entspricht, die der gleiche Fluss in einem unbelasteten Netzwerkelement erhalten würde.

Differentiated-Services Die Fähigkeit eines Router verschiedene Verkehrsklassen im Internet unterschiedlich zu behandeln.
• Die Diffserv-Architektur ist flexibel, da sie keine spezifischen Diente oder Dienstklassen definiert.
⊕ Bei Diffserv-Architektur müssen die Router keinen Pro-Fluss-Zustand verwalten.
▷ Die Diffserv-Architektur setzt sich aus zwei funktionalen Elementen zusammen:

- **Peripheriefunktionen:** An der Peripherie des Netzwerkes werden ankommende Pakete markiert.

- **Weiterleitung**: Wenn ein markiertes Paket an einen Diffserv-fähigen Router ankommt, wird das Paket gemäß dem so genannten *Pro-Hop-Verhalten* in Verbindung mit der Klasse des Pakets an den nächsten Hop weitergeleitet.
  Pro-Hop-Verhalten setzt nicht voraus, dass eine bestimmte Paket-Queuing-Disziplin benutzt wird, um ein bestimmtes Verhalten erreichen.

Neue Architekturkomponenten für Multimedia:

• **Prinzip 1.** Die Paketmarkierung erlaubt es einem Router, Pakte nach unterschiedlichen Verkehrsklassen zu unterscheiden
• **Prinzip 1. (modifiziert)** Die Paketklassifizierung erlaubt es einem Router, zwischen Paketen zu. unterscheiden, die zu unterschiedlichen Verkehrsklassen gehören.
• **Prinzip 2.** Es ist wünschenswert, zwischen mehreren Verkhersflüssen ein gewisses Maß an Isolation bereitzustellen, damit sich ein fehlerhafter Fluss nicht nachteilig auf die übrigen Flüsse auswirken kann.
• **Prinzip 3.** ungeachtet der Bereitstellung zwischen Flüssen ist es wünschenswert, die

Ressourcen (Leitungsbandbreite und Puffer) so effizient wie möglich zu nutzen.
• **Prinzip 4.** Der Prozess der Zugangskontrolle ist notwendig, damit Flüsse ihre QoS-Anforderungen deklarieren und anschließend entweder die angeforderte Dienstqualität erhalten oder vom Netzwerk blockiert werden.

Verbesserungsvorschläge für Multimedia:

• Anwendungen sollten explizit Ende-zu-Ende Bandbreite reservieren.
• Ein Protokoll, das für die Anwendungen Bandbreite von den Sendern zu ihren Empfängern reserviert.
• Scheduling-Strategien in den Router-Warteschlangen zu ändern.
• Mehr Bandbreite, mehr Netzwerk-Caches und umfassende Unterstützung von Multicast.
• Die Anwendungen müssten eine Beschreibung des Verkehrs erstellen.
• Eine Feststellung des Netzwerks, ob es über genügend Bandbreite verfügt, um eine neue Reservierungsanfrage zu unterstützen.
Entfernung von Jitter durch Verwendung von Sequenznummern, Zeitstempeln und einer Wiedergabeverzögerung.